April 1980
$2.50

kilobaud

MICROCOMPUTING T.M.

for business . . . education . . . FUN!

ìreen-Thumb
ìomputing
omputerized
ornucopia. **P. 24**

AP!
auses and
ures of static
lectricity. **P. 54**

lan from C.P.U.
look at
icrocomputing's
eamier side. **P. 102**

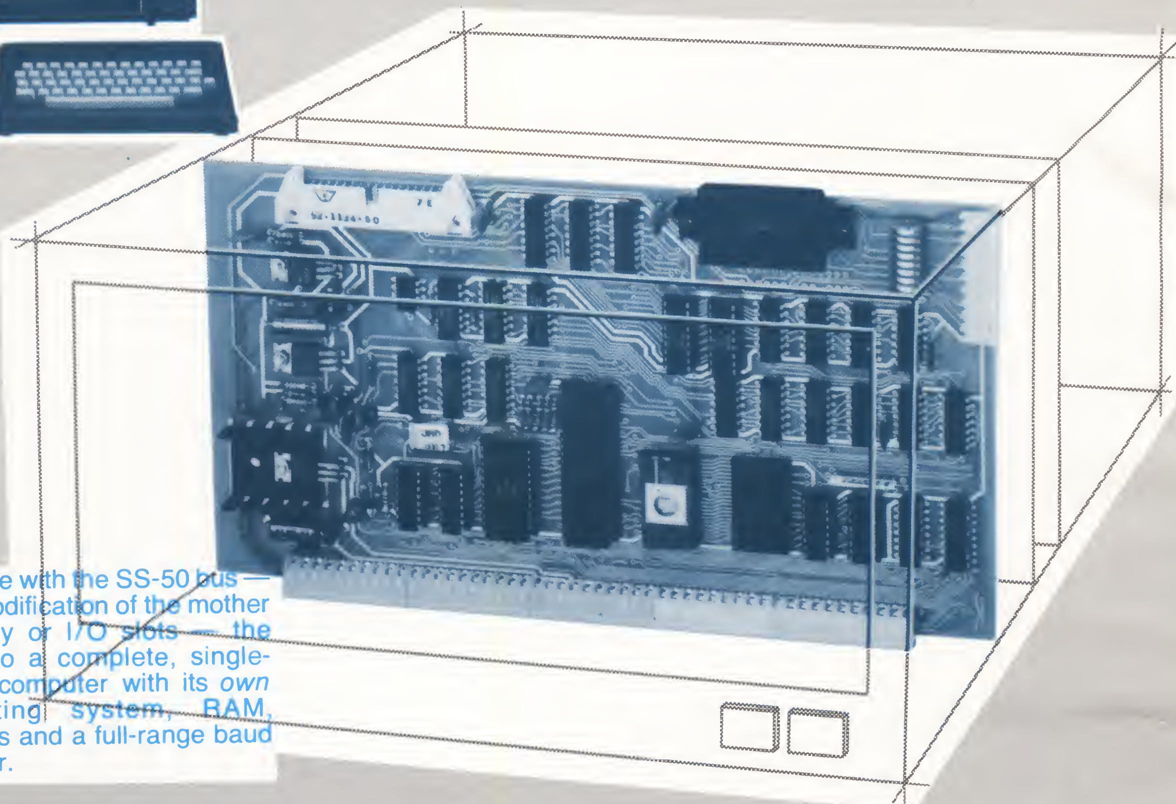
o-All-Plus
BASIC program
r a variety of
ata management tasks.
164

etter Beeper
dd audio
edback to your
omputer keyboard. **P. 200**

04
74470 65945

6809 PROCESSING POWER!

The Percom SBC/9™. Only $199.95.



Fully compatible with the SS-50 bus — requiring *no* modification of the mother board, memory or I/O slots — the SBC/9™ is also a complete, single-board control computer with its *own* ROM operating system, RAM, peripheral ports and a full-range baud clock generator.

Make the SBC/9™ the heart of your computer and put to work the most outstanding microprocessor available, the 6809.

the Mighty 6809

Featuring more addressing modes than any other eight-bit processor, position-independent coding, special 16-bit instructions, efficient argument-passing calls, autoincrement/autodecrement and more, it's no wonder the 6809 has been called the "programmers dream machine."

Moreover, with the 6809 you get a microprocessor whose programs typically use only one-half to two-thirds as much RAM space as required for 6800 systems, and run faster besides.

And to complement the extraordinary 6809, the Percom design team has developed PSYMON™, an extraordinary 6809 operating system for the SBC/9™.

PSYMON™ — Percom SYstem MONitor

Although PSYMON™ includes a full complement of operating system commands and 15 externally callable utilities, what really sets PSYMON™ apart is its easy hardware adaptability and command extensibility.

For hardware interfacing, you merely use simple, specific device driver routines that reference a table of parameters called a Device Control Block (DCB). Using this technique, interfacing routines are independent of the operating system.

The basic PSYMON™ command repertoire may be readily enhanced or modified. When PSYMON™ first receives system control, it initializes its RAM area, configures its console and then 'looks ahead' for an optional second ROM which you install in a socket provided on the SBC/9™ card. This ROM contains your own routines that may alter PSYMON™ pointers and either subtly or radically modify the PSYMON™ command set. If a second ROM is not installed, control returns immediately to PSYMON™

- Provision for multi-address, 8-bit bidirectional parallel I/O data lines for interfacing to devices such as an encoded keyboard.
- A serial interface Reader Control output for a cassette, tape punch/reader or similar device.
- An intelligent data bus: multi-level data bus decoding that allows multiprocessing and bus multiplexing of other bus masters.
- Extended address line capability — accommodating up to 16 megabytes of memory — that does not disable the on-board baud rate clock or require additional hardware in I/O slots.
- On-board devices which are fully decoded so that off-card devices may use adjoining memory space.
- Fully buffered address, control and data lines.

The SBC/9™, complete with PSYMON™ in ROM, 1K of RAM and a comprehensive users manual™ costs just $199.95.

™ trademark of Percom Data Company, Inc.

PERCOM

PERCOM DATA COMPANY, INC.
211 N. KIRBY GARLAND, TEXAS 75042
(214) 272-3421

Percom *'peripherals for personal computing'*

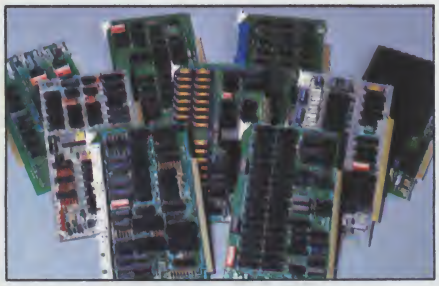
✓1

To place an order or request additional literature call toll-free 1-800-527-1592. For technical information call (214) 272-3421. Orders may be paid by check, money order, COD or charged to a VISA or Master Charge account. Texas residents must add 5% sales tax.

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE

At Intersystems, "dump" is an instruction. Not a way of life.

(Or, when you're ready for IEEE S-100, will your computer be ready for you?)



We're about to be gadflies again.

While everyone's been busy trying to convince you that large buses housed in strong metal boxes will guarantee versatility and ward off obsolescence, we've been busy with something better. Solving the *real* problem with the first line of computer products *built from the ground up to conform to the new IEEE S-100 Bus Standard*. Offering you extra versatility in 8-bit applications today. And a full 16 bits tomorrow.

We call our new line Series II.™ And even if you don't need the full 24-bit address for up to 16 megabytes (!) of memory right now, they're something to think about. Because of all the performance, flexibility and economy they offer. Whether you're looking at a new mainframe, expanding your present one or upgrading your system with an eye to the future. (Series II boards are compatible with most existing S-100 systems and *all* IEEE S-100 Standard cards as other manufacturers get around to building them.)

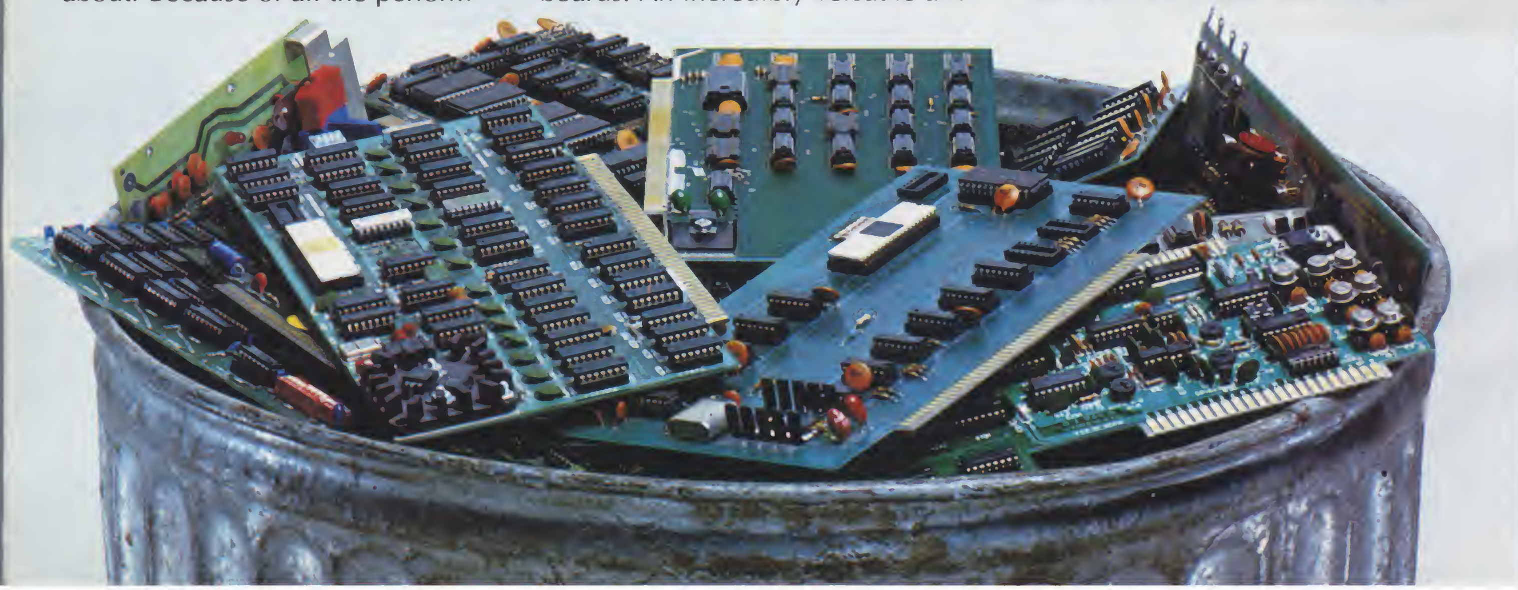
Consider some of the features: Reliable operation to 4MHz and beyond. Full compatibility with 8- and 16-bit CPUs, peripherals and other devices. *Eight* levels of prioritized interrupts. Up to 16 individually-addressable DMA devices, with IEEE Standard overlapped operation. User-selectable functions addressed by DIP-switch or jumpers, eliminating soldering. And that's just for openers.

The best part is that all this heady stuff is available *now*! In our advanced processor—a full IEEE Bus Master featuring Memory Map™ addressing to a full megabyte. Our fast, flexible 16K Static RAM and 64K Dynamic RAM boards. An incredibly versatile and economical 2-serial, 4-parallel Multiple I/O board. 8-bit A/D-D/A converter. Our Double-Density High-Speed Disk Controller. And what is undoubtedly the most flexible front panel in the business. Everything you need for a complete IEEE S-100 system. Available separately, or all together in our new DPS-1 Mainframe!

Whatever your needs, why dump your money into obsolete products labelled "IEEE timing compatible" or other words people use to make up for a lack of product. See the future now, at your Intersystems dealer or call/write for our new catalog. We'll tell you all about Series II and the new IEEE S-100 Bus we helped pioneer. Because it doesn't make sense to buy yesterday's products when tomorrow's are already here.

InterSystems™

Ithaca Intersystems Inc.,
1650 Hanshaw Road/P.O. Box 91,
Ithaca, NY 14850
607-257-0190/TWX: 510 255 4346



kilobaud

MICROCOMPUTING™

contents: apr. '80

ARTICLES

DEPARTMENTS

Cover: Photographed by Reese Fowler; consumed by *Microcomputing* staff.

micro info

$ This symbol next to a title in the table of contents indicates that the article is a business-application article.

Manuscripts

Contributions in the form of manuscripts with drawings and/or photographs are welcome and will be considered for possible publication. We can assume no responsibility for loss or damage to any material. Please enclose a self-addressed, stamped envelope with each submission. Payment for the use of any unsolicited material will be made upon acceptance. All contributions should be directed to the *Microcomputing* editorial offices. "How to Write for Microcomputing" guidelines are available upon request.

Editorial Offices:

Pine Street
Peterborough NH 03458
Phone: 603-924-3873, 924-3874

Advertising Offices:

Pine Street
Peterborough NH 03458
Phone: 603-924-7138, 924-7139

Circulation Offices:

Pine Street
Peterborough NH 03458
Phone: 603-924-7296

To subscribe, renew or change an address:

Write to *Microcomputing*, Subscription Department, PO Box 997, Farmingdale NY 11737. For renewals and changes of address, include the address label from your most recent issue of *Microcomputing*. For gift subscriptions, include your name and address as well as those of gift recipients. Postmaster: Send form #3579 to *Microcomputing*, Subscription Services, PO Box 997, Farmingdale NY 11737.

Subscription problem or question:

Write to *Microcomputing*, Subscription Department, PO Box 997, Farmingdale NY 11737. Please include an address label.

Kilobaud *Microcomputing* (ISSN 0192-4575) is published monthly by 1001001, Inc., 80 Pine St., Peterborough NH 03458. Subscription rates in U.S. are $18 for one year and $45 for three years. In Canada: $20 for one year and $51 for three years. In Europe, send 89,-DM in Eurocheque or send credit card information to: Monika Nedela, Markstr. 3, D-7778 Markdorf, W. Germany. South African Distributor: KB Microcomputing, PO Box 782815, Sandton, South Africa 2146. All other foreign subscriptions are $23—one year only (surface mail). Second-class postage paid at Peterborough NH 03458 and at additional mailing offices. Phone: 603-924-3873.

Wayne Green

PUBLISHER'S REMARKS

Impact of "80"

The success of *80 Microcomputing* has been gratifying, particularly in contrast to the failure of other new microcomputer magazines in recent months. We printed 50,000 of the first issue, and those are rapidly disappearing as dealers order and reorder.

There were two reasons for starting the new magazine. One was the growing amount of information about the TRS-80 which needed to be published. *Microcomputing* was publishing more TRS-80 material than the other major magazines, and still was not keeping up. It looked as if the TRS-80 might push all the other systems out of the magazine. Now, with "80," it is possible for *Microcomputing* to concentrate more on the Apple, PET, Heath and other fine microcomputer systems. This will benefit the industry and the readers.

The other reason was keeping ad costs reasonable. High ad costs keep many new and small firms out of the business. Splitting the coverage of the market into two magazines allows us to keep ad rates for each low enough to encourage new firms to start and flourish.

What does this mean to you, the reader? It means you'll find a wider selection of new and innovative products advertised in *Microcomputing*. The lower-cost advertising means you'll have a wider range of products from which to select, many available only on a mail-order basis until their volume grows to where dealers can handle the product with confidence . . . and at a profit.

In terms of getting interesting reading material, what does the split mean to subscribers? A rough count of the number of article pages shows the January issue of *Microcomputing* with 94 pages, the January issue of *80 Microcomputing* with 71 pages and the January issue of *Byte* with 85 pages. The reader of *Microcomputing* and "80" ends up with 165 pages of articles, almost double the number the *Byte* reader gets, and *Byte* tends to ignore the TRS-80. So does *onComputing, Byte's* new publication.

It Is Still a Hobby

Despite persistent rumors that IBM will soon announce a microcomputer system, I am not surprised by delays. If I were the sales manager of a major firm about to enter the microcomputer market I would have qualms about jumping in right now.

First, I would evaluate the competition by getting a TRS-80 and the available software for it . . . same for the Apple and perhaps a couple others. I would check out the available systems from the viewpoint of a businessman. My report would conclude that there is little of value available for business so far.

I'd also read over the magazines published for the field and see what business systems they'd reviewed. My finding virtually nothing would tell me something.

If the largest hardware firms in the business still have been unable to generate any reliable and useful business software, obviously the sales so far, no matter what dealers say, have been mostly to hobbyists or people who have been given a con job.

I've seen the reports in magazines that some dealers are selling 50 percent of their systems for business purposes. If this is true, let's see some reports from *users.*

Having worked with Instant Software for almost two years, I know what it takes to sort through already written software and publish it. I don't think *any* systems manufacturers have the people and facilities ISI has, so I doubt if any of them can produce as much good software. The day will come when the public realizes that no hardware can do anything unless it is supported by a lot of software.

Looking to Buy

If you know of any firms or dealers who are going out of business, let them know that we want to hear from them. With our lab expanding at such a rapid pace, we need a lot of equipment. I'd like a chance to put in a bid for the whole works: lock, stock and barrel. This can save a lot of time and expense for a firm that wants to liquidate.

We can use just about any hardware—complete systems, memories, chips, I/O boards, printers, terminals, modems—and software of almost any kind. We need test equipment, disks, tapes, instruction books, publications . . . you name it.

Our laboratory is, I believe, the best equipped microcomputer lab in the world, and we want it to be even better. We want to be able to check out and report on any software . . . to be able to check and use any accessories. This is helpful to manufacturers and the programmers, and it's valuable to every reader. The better informed we are, the better the magazine will be.

Maryland Computerfest

The only Eastern computerfest at which I am scheduled to give a talk this year is in the Baltimore area on March 30th at the Maryland State Fairgrounds in Timonium. This combination computerfest/hamfest will feature computer and ham exhibits and a large flea market, plus prizes and talks.

I'm really looking forward to the opportunity to get together with readers in the Washington-Baltimore area, answer questions and tell you where I think things are going (and how to take advantage of it). I'll be interested in talking about what you like or would like to see in *Kilobaud* and "80," and bring you up to date on the action at Instant Software.

Prime Troubles . . . Again!

Again I have to offer apologies to readers who have been inconvenienced by problems with our Prime computer. This time it has to do with Reader Service requests for the last three months. We'll have a new system up and running for the April issue, but service problems with the Prime have delayed earlier responses. A recent letter to the president of Prime asking for help with our Prime problems was answered, not by their service department, but by their legal department.

The problems with the Prime have forced us to put our repeater lists on a TRS-80 system so we can update the *73 Magazine* list of the world's repeater stations. We are removing even the smaller jobs from the Prime and doing them on TRS-80 systems or a Midwest Scientific Instruments microcomputer system. So far the TRS-80 has been far more dependable than the Prime, and when we have problems, Radio Shack is cooperative.

Readers will remember the monumental snafu when we tried to use the Prime for handling subscriptions, and the whole system ground to a halt. We made tens of thousands of readers mad at that time and lost a great deal of subscription income as a result. This, in turn, resulted in a substantial loss of advertising revenue. We not only learned an expensive lesson, but we have the material for a horror story of how computers can be an expensive disaster when they fail in their application.

Sherry Smythe

OUTPUT FROM ISI

State of the Art

Most programmers are keeping one eye on equipment sales, so it is not surprising that more software is being written for the TRS-80 Model I. Radio Shack is probably selling no less than all other systems combined. If Radio Shack becomes complacent, other systems may be able to counter the might of Radio Shack merchandising with better hardware, better software support and better marketing.

The Model II TRS-80 is a case in point. Profits on Model IIs aren't attractive; some of the hardware is months back-ordered and software support is weak. Model II is not now a good bet for use in software development on a free-lance basis. That will come when there are more users and some accessories provided by other manufacturers.

One manufacturer is about to release a translator program that will enable programmers to convert TRS-80 software so it can be used on his system. The translation will be about 98 percent complete, leaving programmers with a few changes to make by hand. I suggest that any programmers capable of handling this type of software write translators. These should sell well.

Keep your eyes open for manufacturers who plan to take on Radio Shack. It isn't going to be easy. Although the Shack does give dealers a short markup on computers, the company supports them with help at shows and with national advertising.

The key to any long-term increase in sales lies in software support. So far, no manufacturer has made a serious effort to provide much software. Some day a firm is going to tackle the software problem and lay the foundation for success.

In looking over the best-selling ISI programs for January, I find the Flight Simulator (0017R) way out in front. In second place is 0081R, Utility I. In third place is a new package, 0106R, Airmail Pilot. Next is 0076R, Utility II, followed by 0034R, Trek IV, and then 0103R, Personal Bill Paying, another new package. All are TRS-80 programs.

The best-selling Apple program was 0018A, Golf. This was followed by 0073A and 0098A, the two math-tutor programs.

Trek-X, 0032P, was the best-selling PET program, just nudging out the top-selling Apple program and having about one-third the sales of the top Radio Shack program. In second place was Dungeon of Death, 0064P; third was Dow Jones, 0026P. Despite all the talk about the need for business programs, game programs continue to be the best-sellers. There may be a message there.

BOOK REVIEWS

How to Make Money with Your Microcomputer

Merl Miller, Carl Townsend
Robotics Press, Forest Grove OR
Softcover, 152 pages, $6.95

People are discovering the personal-computing hobby. It is only natural that some of these hobbyists try to turn their computers into money-makers. Is it possible to earn a living with a microcomputer? In *How to Make Money with Your Microcomputer*, Miller and Townsend point out some of the ways to do it.

Twelve detailed descriptions of possible computer businesses are given. These range from writing magazine articles to opening your own computer store. They include running a service bureau, creating and selling hardware or software, writing books, putting on computer shows or conventions, becoming a consultant and starting a computer repair business.

All of these suggestions are backed up with both pros and cons. The final decision is up to you, but this information may help clarify your thinking. Establishing, financing and managing a business are discussed, as are marketing techniques, which may be the most important component of a successful business. Examples given in each chapter make the discussion easier to understand. You won't be able to start a business and get rich just from reading this book, but it could start you thinking.

Chapter 1 covers writing articles for microcomputer magazines. The basic information

kilobaud
MICROCOMPUTING T.M.

PUBLISHER/EDITOR
Wayne Green
EXECUTIVE VICE PRESIDENT
Sherry Smythe
CORPORATE CONTROLLER
Alan Thulander
ASSISTANT PUBLISHER/EDITOR
Jeffrey DeTray
MANAGING EDITOR
John Barry
EDITORIAL ASSISTANTS
Dennis Brisson
Susan Gross
ADMINISTRATIVE ASSISTANT
Leatrice O'Neal
PRODUCTION DEPARTMENT
MANAGER:
Noel Self
ASSISTANT MANAGER:
Robin Sloan
STAFF:
Steve Baldwin
James Butler
Robert Drew
Bruce Hedin
Kenneth Jackson
Clare McCarthy
Michael Murphy
Dion Owens
Nancy Salmon
Patrice Scribner
Susan Symonds
John White
TYPESETTING
Barbara Latti
Sara Bedell
Sandie Gunseth
Mary Kinzel
Karen Podzycki
PHOTOGRAPHY
William Heydolph
Tedd Cluff
Terrie Anderson
Reese Fowler
PROJECTS EDITOR
Jim Perry
BOOK EDITORS
Chris Brown
Emily Gibbs
ASSOCIATE EDITORS
Frank Derfler, Jr.
Rod Hallen
Peter Stark
Sherm Wantz
ACCOUNTING MANAGER
Knud Keller
CIRCULATION MANAGER
Debra Boudrieau
CIRCULATION
Barbara Block
Pauline Johnstone
BULK SALES MANAGER
Ginnie Boudrieau
EUROPEAN MARKETING DIR.
Reinhard Nedela
ADVERTISING
Aline Coutu, Mgr.
Marcia Stone
Penny Brooks
Nancy Ciampa
Louise Holdsworth
Jerry Merrifield
Rita Rivard
Kevin Rushalko
Hal Stephens
Phoebe Taylor

actually pertains to writing articles on any subject for any magazine. First you have to find a market and determine what kind of articles interest the market's readers. Then follows a discussion of actually writing the article. Such topics as outlines, drafts, final copies and illustrations are covered in detail. Different types of articles—how-to, tutorial, hardware, software, book reviews and others—are discussed. You will undoubtedly want to use your computer as a word processor to speed up and improve the creation and quality of your manuscript. Chapter 2 extends this discussion.

Not all these ideas will appeal or apply to you. Your temperament, background, education and other factors rule some of them out. On the other hand, one or more of the suggestions in this book may point your thinking in some other direction. The microcomputer business is growing rapidly, and no one knows exactly which direction it will take. However, there will be thousands of job and income opportunities available in the next few years. *How to Make Money with Your Microcomputer* might just get you started in one of them.

Rod Hallen
State Dept.

Digital Image Processing

Kenneth R. Castleman
Prentice-Hall, Inc.
Englewood Cliffs NJ, 1979, $25

Ken Castleman's book explains computer image processing. While intended as a text aimed at graduate students of computer science, two-thirds of the text (sections I and III) is understandable to computer hobbyists, programmers or engineers. Even if you are put off by Fourier transforms or double integrals, there is still enough of interest in this book to keep you busy dreaming of the things you can make your own little computer accomplish . . . with a little help.

It is section II that is heavy in the math department. You can skim or skip this section and accept the remainder of the text as an overview of what is, after all, a math-intensive activity. Taken as an overview, the rest of the book is still interesting, rewarding and understandable, thanks to the author's skill in presenting topics in both word descriptions and equations. If you want some action as well as understanding, though, you'll need some hardware help in addition to a copy of this book.

The first thing you'll need is a video display board capable of displaying at least 256 by 256 pixels of four bits (sixteen levels) of intensity and/or color. A few years ago this would not have been within the means of even the most dedicated hobbyist, but today such hardware is available from Matrox and Vector Graphic, to name a couple. However, since this involves an investment of over a thousand dollars just for display hardware, image processing is still for the serious experimenter, computer club or professional.

But understanding the magic of image processing operations is within the reach of all of us, thanks to this book. In terms understandable to the non-mathematician, Dr. Castleman explains the processes used to provide breathtaking enhanced views of the other planets and their satellites.

Operations such as the elimination of background noise, the enhancement of contrast and the detection and enhancement of object edges are fully explained in this text. Since most of the examples used are in the realm of spacecraft camera image processing, you might wonder what good the knowledge is for the computer hobbyist. Some of the techniques discussed can be used by individuals interested in photography or slow-scan TV.

With more processing power, as will soon be available in the 16-bit micros, we can move on to the more exotic aspects of image processing: rotations, distortion corrections and such. Within the confines of the hardware and knowledge available to the serious experimenter or computer club is a wealth of interesting and potentially rewarding activities in the realm of image processing that are within reach of small computers. Castleman's text is not a how-to-do-it book. For the hobbyist/experimenter it is a dream book filled with intriguing ideas.

Ken Barbier
Borrego Springs CA

Introduction to Microprocessor System Design

Harry Garland
McGraw-Hill, New York, 1979
Softcover, 192 pages, $10.95

Here is another book on microcomputers whose title leads you to believe it can take you from "nowhere" to "somewhere" in a few easy lessons. Yet its preface assumes a background of "a one-year undergraduate electronics course or equivalent" and "an introduction to computer programming, a familiarity with the concept of a stored program and a knowledge of the binary number system." But even with this background under your belt, the "somewhere" you might reasonably expect to reach is not very far.

In chapter 7, "Assembly and High-Level Languages," for example, assembly language is "covered" in four pages, and PL/M and BASIC each get two pages! The book does allocate 24 pages to chapter 6, "Microprocessor Machine Language," but much of this is "filler": two-page ASCII table and a seven-page listing of the binary instruction codes for the Z-80/8080/8085 microprocessors. The other 15 pages "cover" machine-language programming, machine cycles, registers, addressing modes, stack operations and subroutines.

Even though the book does not deliver all that its title implies, it is not all bad. If it had been titled "Introduction to the 8008 Microprocessor and Its Descendants" or perhaps "Intel and Zilog Microcomputer Hardware Systems," or the like, then the contents would have matched the title. The present format and arrangement of the material are good.

In addition to the two rather useless chapters already mentioned, the other chapters include:

1. "Introduction," which begins with a good discussion of an *ideal* microprocessor before tackling the nitty-gritty details of real-world chips.

2. "Microprocessor Technology," a discussion of basic transistor circuitry and of bipolar and MOS technology.

3. "Microprocessor Evolution," which describes the 8008, 8080, Z-80, 8748, 8086 and Z8000 microprocessors.

4 and 5. "Basic Microprocessor Hardware" and "Expanding the Microprocessor System." These are the two "meaty" sections of the book, composed mostly of portions of nifty little circuits that help make a microprocessor chip function as it should. These include 3-terminal voltage regulators, clock generators, address decoders, status latches, wait-state generators, single-steppers and DMAs. Unfortunately, most of this information seems to be copied from various manufacturers' data sheets, with a couple of pictures of commercially available PC boards thrown in for good measure. Both pictures are from Cromemco Incorporated, of which the author is president.

8. "Microprocessor Arithmetic": addition, subtraction, multiplication and division in the binary number system. Division is "covered" in one six-line paragraph, and two exercises are left for the reader!

9. "Analog Interfaces": D/A and A/D, again featuring mostly Cromemco equipment and photographs.

10. "Interface Standards," essentially a description of the S-100 and IEEE 488 buses and the RS-232 communications interface. Again, several pictures of Cromemco equipment are included.

In addition to the circuits, the other positive feature that impressed me was the section in the back of the book which answered the odd-numbered problems at the end of each chapter. Each chapter had about ten exercises; by performing them and then checking the answers, the reader who is using the book in a non-classroom situation can at least determine if he/she is learning anything.

This book appears to have been "cranked out" by a university professor who was under the Damoclean sword of "publish or perish." Why McGraw-Hill, which has numerous good books to its credit, would accept a book of this sort is beyond my comprehension.

Myron Calhoun
Manhattan KS

Microprocessor Applications in Business and Industry

Marvin Whitbread, Ed.
Castle House Publications Ltd.
Kent England
Softcover, 153 pages, £9.50

Microcomputer Applications is the first volume of a projected series called "Topics in Microprocessing." It is a collection of English-language articles reprinted from British and American magazines. The editor was formerly with the Microprocessor Project of the U.K. National Computing Centre.

The book is divided into five sections. The first section presents four introductory articles; together they form a primer on software con-

cepts, hardware concepts and the structure of the microprocessor industry. One of the articles, reprinted from *The Economist*, also includes an interesting explanation of integrated-circuit fabrication.

Section two is devoted to business applications. Of the eight articles in this section, among the most revealing (and among the most misplaced) is "Owner's Report—The TRS-80," a British-eye view of the ubiquitous Radio Shack machine. This particular piece is "Verrie Britishe" throughout, both in diction and attitude. Of the Radio Shack Level I BASIC manual, it complains that it "is written in a folksy, 'let's you and me and the computer be friends together' style, common in the U.S. but which will surely grate on the nerves of some customers over here."

Section three describes industrial applications of microcomputers. Most of the articles are case studies of factories using computer-controlled assembly equipment—in other words, industrial robots. Although this section is not directly useful to hobbyists and small businessmen, the technology involved makes stimulating reading.

Section four covers miscellaneous applications ranging from medical care to automobiles. I question the relevance of one of editor Whitbread's selections here: a *Creative Computing* article about the 1978 Toy Fair. It is nothing more than a series of short descriptions of new toys and games—both electronic and non-electronic. The information may be useful, but is this book really the place for it?

Section five is entitled "Microprocessors and Management." The five articles in this section deal with varied topics, such as microprocessor design techniques, the trade-union view of microelectronic automation, and the long-term effects of microprocessor proliferation. (The latter article is peppered with some wonderful sociologist-style buzztalk.)

The curious novice will find *Microprocessor Applications* interesting, as long as he does not expect meaty information. The articles are broad treatments of their respective topics. Whether this is an asset or a liability depends on your level of expertise.

David Price
Midlothian VA

Robert W. Baker

PET-POURRI

Poking Around in BASIC

Normally you don't have to know anything about the internal workings of BASIC in the PET. There are times, however, when it may help to know even the simplest details. Many articles on how BASIC lines are stored in memory have appeared, and the format is illustrated in the detailed PET memory map of the PET user manual. Briefly, each BASIC line has a five-byte overhead when stored in memory. Four bytes precede the line of text, and a single byte follows the line with a value of zero to indicate the end of the line. See Fig. 1.

The four bytes at the start of each line contain a two-byte link, which is an address pointer to the starting location of the next line, and the BASIC line number stored as a two-byte binary number. The link and line number are both stored in standard 6502 address format. This means that the first byte is the low order eight bits of the address and the second byte contains the high order eight bits of the total 16-bit binary value. The end of the BASIC program is indicated by a link with both bytes equal to zero. The actual BASIC text is compressed, with all BASIC statements and commands stored as single byte "tokens" to conserve memory space.

With this information in hand, I experimented to see how BASIC in the PET used these five overhead bytes for the link, line number and end-of-line flag during various functions or commands. I first tried poking a single zero byte into the middle of a BASIC line that was already stored in memory, and then listed the program. The entire program was still listed except for the characters in the one line that was modified following the new zero byte.

I then tried poking three sequential zero bytes into the middle of a BASIC line to simulate an end-of-line flag and a zero link in the middle of a program. This had the same effect when the program was listed again; the entire program, except for the end of the modified line, was listed.

This proves the LIST command uses the link information to go from one line to the next and displays each line till finding the end-of-line flag, a single zero byte. Furthermore, the LIST command does not check that the link points to the next byte after the end-of-line flag; it assumes the link is correct.

Next I ran a program with a single zero byte poked into the middle of various lines that contained remarks or executable statements. When the modified line executed, it caused the program to fail with unpredictable results. If the modified line did not execute by branching around it, the program ran perfectly.

I then tried poking three sequential zero bytes into the middle of various lines simulating an end-of-line flag and a zero link in the middle of a program. When the modified line executed, the program terminated at that point as if that was the normal end of the program. If the modified line did not execute, the program ran to the normal end of the program. This showed that BASIC used the link values to find a given line whenever lines were not executed sequentially. However, all links are ignored and not verified whenever BASIC lines are executed sequentially.

Further testing following similar lines proved that a program was saved on tape until the first three sequential zero bytes were found, regardless of where these bytes occurred. A program SAVE is a straight "memory dump," storing each consecutive byte of memory from the start of the program until three consecutive zero bytes are found. Saving and then loading a program that has a single zero byte poked into the middle of a BASIC line produces some strange results.

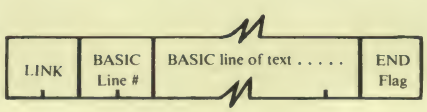


Fig. 1.

Apparently, all the link values of the BASIC program are corrected after the program is loaded. BASIC "knows" the end of each line when it finds the end-of-line flag, the single zero byte. Thus, the data in the modified line following the added zero byte is interpreted as an extra line that may produce garbage with a strange line number in the middle of your program. Trying to edit program lines when extra zero bytes have been poked into the program can cause similar results when the edit routine tries to relink the BASIC program.

By trying various ways to change the line numbers in a BASIC program, I found that BASIC used the link values to search through the program whenever looking for a particular line. If you find a line number in the program that is greater than the number you're searching for, the search ends unsuccessfully. When two lines have the same line number, the one closest to the start of the program is always used since it is found first in the search. These rules are used for all functions that require locating a particular BASIC line, such as GOSUB, GOTO, RUN xxxx, LIST xxxxx or screen editing.

BASIC will not allow entering any line with a number greater than 63999. However, you can poke a new value to change an existing line number to any value greater than this limit, up to 65535. The line will still list and run correctly, but cannot be edited or deleted from the program since the line number is invalid. A line that is to be "protected" in this manner should be located at the end of a program since any fol-

(continued on page 14)

OHIO SCIENTIFIC'S

SMALL SYSTEMS JOURNAL

Ohio Scientific Multiple User Systems

This is the continuation of last month's Small Systems Journal concerning the Ohio Scientific multiple user systems.

The previous portion of this article covered primarily Ohio Scientific's timesharing systems. In the following pages, the hardware and software used in Ohio Scientific's networked computer systems will be covered.

Introduction

The Network Extension to OS-65U provides the capability to interconnect up to sixteen hard disk based Level 3 timesharing systems. Also, up to fifteen floppy disk based 65U Level 2 or C2-NET systems can be connected to each Level 3 system. And each of these can further support up to sixteen Level 1 intelligent terminals. In all, literally thousands of users can be interconnected for hard disk data and program sharing with the OS-65U Network Extension.

The speed of data transfer between computers can be as high as 500,000 bits per second yielding nearly immediate response to transfer requests. Level 2 users' transfers to or from Level 3 systems to which they are indirectly connected occur with only a slight additional delay.

Accessing remote data bases with the OS-65U Network Extension is as easy as accessing a standard disk memory device. Under OS-65U, the letter designations A, B, C and D refer to floppy disks and E refers to a hard disk. Under the Network Extension the *local* hard disk can always be referenced as device E and any of the other hard disks may be referenced by the absolute designations K through Z. These device designations may be used in the 65U 'DEV' statement just as would any disk designation. Thus, upgrading existing application software to networking requires relatively few, if any, changes.

For those applications using shared data files, the Level 3 semaphore commands WAIT FOR and WAIT CLEAR are automatically extended to affect semaphores located at the referenced data base. For example, a user who will be accessing a shared file on data base Z can coordinate his file accesses with other users by executing a 'WAIT FOR n' command after specifying DEV"Z". All users performing the same steps will reference the specified semaphore located at data base Z thereby insuring that the file accesses are executed in an orderly manner.

Also, to better suit application needs, all networked systems permit the programmer to specify a time limit for WAIT FOR commands, then check the results of the operation after the command has been executed. This permits giving the computer operator the option of continuing to wait for a busy file or postponing his request. It also provides an easily used mechanism for safely acquiring multiple shared resources. Yet another feature of the Network Extension automatically clears all user locked semaphores when a CLEAR or RUN command is executed or the direct (command) mode of BASIC is entered.

In summary, the OS-65U Network Extension provides the many benefits afforded by distributed processing to Ohio Scientific microcomputer users with an ease of use and implementation that makes it the logical choice to satisfy expanding data processing needs.

Operation

Network System Startup

The OS-65U network is started up by booting all of the interconnected computer systems to be used and running the 'LEVEL3' program at each hard disk based network "node" to bring the timesharing systems on-line.

Then one of the memory partitions in each timesharing system is selected as the network support partition. At the terminal associated with this partition the network support program 'NETWRK' is run. This program reports the unique node designation (K through Z) assigned to the node and awaits network messages. At this time the terminal for this partition ceases to accept keyboard input. It becomes a network monitor, reporting message traffic and any errors that might occur.

Next, a special program is run at each user terminal that is to be given access to the network.

Timesharing users or Level 2 users that do not first run the appropriate network interface program from their terminal do not have access to the network. Thus, selective distribution of these programs can be used to control network access even at user locations which are physically wired to the network.

Specifying User Access Permission

Remote users' access to the node data base can be controlled at each node. This is done by specifying unique hard disk access limits for each Level 2 user and for the other network users accessing the node data base via other nodes. The initial access limits (when the node is booted) give all users access to the first two cylinders of the hard disk data base. This is addresses 0-430,079 on a CD-74 disk and 0-229,375 on a CD-23 disk.

The CONSOL program is used to change the access limits of a network user.

Accessing the Network

As described previously under "Network System Startup", both timesharing and Level 2 users must run the appropriate network interface program (TSNET or L2NET, respectively) before they have access to the network. After having run this program they can access any programs or data on any hard disk data base in the network within the range they have access permission.

To access a given hard disk data base the user merely executes the OS-65U DEV"a" command, specifying for "a" the letter designation of the desired hard disk based node. Level 2 users (and, of course, timesharing users) can access the "local" hard disk as device E. (The "local" hard disk for a Level 2 user is the hard disk located at the node to which his computer is physically wired.) Other hard disks in the network can be accessed by using the letter designation K through Z as was assigned to each node.

After specifying the desired node by executing a 'DEV' command, all 'RUN', 'LOAD' and 'SAVE' commands executed will access programs at the selected remote node. Likewise, the 'OPEN', 'PRINT%n' and 'INPUT%n' commands can access data files at the remote node. With a 500,000 bits/second network transfer rate, response to these commands is very little slower than accesses to a directly connected disk.

The standard OS-65U utility programs provided on the Network diskette will also operate on a remote data base. To do so the ap-

propriate node designator is entered in response to the "UNIT?" question from the program. Those utilities which alter the file directory (CREATE, DELETE and RENAME) will lock the directory semaphore at the remote node to prevent a conflicting access by another network user. If the directory semaphore is already locked when the program attempts to lock it the following message is output:

DEVICE a DIRECTORY IS BUSY
PLEASE TRY AGAIN LATER

Wait a few seconds and try again.

Using Semaphores to Coordinate Shared File Access

Under network operation the Level 3 semaphore commands WAIT FOR and WAIT CLEAR are automatically extended to reference semaphores located at the hard disk node specified by the last DEV"a" command. As described in the previous article, the WAIT commands are used to coordinate access to shared files when more than one user might attempt to alter the file at the same time.

As an example of a shared file access consider an inventory file that contains quantities of items in stock. As parts are received the quantities are incremented by a network user in the Receiving Department. The quantities are decremented by another user in the Shipping Department when items are shipped from stock. Each user's access to the inventory file must be coordinated with the other users' access, or an incorrect file update can occur. An example of how this can happen is as follows:

Receiving Department	**Shipping Department**
1. Reads the quantity in stock of an item, say it's 3.	Reads the quantity in stock of the same item, i.e., 3.
2. Increments quantity to 4.	Decrements quantity to 2.
3. Writes revised quantity back as 4.	
	4. Writes the revised quantity back as 2.

Since the Shipping Department was last to write the revised quantity back to the inventory file, it now shows a quantity of 2. Had the Receiving Department been last, the quantity would be 4. Of course, neither is correct.

File accesses are coordinated to prevent the above problem by using the WAIT commands to manipulate a semaphore located at the node containing the inventory file. If that were node "M" and the agreed upon semaphore number for the inventory file "resource" were 15, the above scenario would take place as follows:

Receiving Department	**Shipping Department**
1. Executes WAIT FOR 15 which locks resource 15.	Executes WAIT FOR 15 which suspends execution because resource 15 is locked.
2. Reads the quantity in stock of the item, 3.	
3. Increments quantity to 4.	
4. Writes revised quantity back, 4.	
5. Executes WAIT CLEAR 15 unlocking resource 15.	
	6. Continues execution with resource 15 locked on its behalf.
	7. Reads the quantity in stock, 4.
	8. Decrements quantity to 3.
	9. Writes revised quantity back, 3.
	10. Executes WAIT CLEAR 15 unlocking resource 15.

Thus, the use of the WAIT FOR and WAIT CLEAR commands ensures the proper update of the file.

In the above example a semaphore was used to lock the whole inventory file during an update. Under some circumstances it may be desirable to lock only one record in a file when that record is to be updated so as to leave the remainder of the file accessible to other users for updates.

Record level locks can be used in OS-65U if the following criteria are met:

—File boundaries must fall on sector boundaries. (Standard under Level 3.)

—Each shared file record must have a two character record lock field.

—File readers must be aware that if they read a record without locking the semaphore and find that the record is locked the data within that record may be inconsistent. (This can occur on a record that overlaps a sector boundary if the record is read between the times when one sector of the record was written and when the other sector of the record is written.)

—File writers must PRINT to the record lock field after updating all other fields within the record.

—Each user of a shared file uses this procedure to change a record:

(**NOTE:** 1* refers to the number of a semaphore that all users of the shared file agree to use to coordinate their accesses to the file's record locks.)

```
n WAIT FOR 1*             Wait for record locking permission.
  OPEN"filename",n        Got it—Open desired file.
  INDEX[n] = . . .        Set Index to desired record.
  INPUT%n,L$              Input record lock field.
  IF L$ = "0" GOTO m      If not locked continue.
  CLOSE n                 Else close file,
  WAIT CLEAR 1*             give up record lock permission
  GOTO n                    and go try again.
m INDEX[n] = . . .        Reset Index to desired record.
  L$"1":PRINT%n,L$        Set record lock to prevent others
  CLOSE n                   from accessing the record.
  WAIT CLEAR 1*           Close file and permit others to lock.
```

Now the desired file may be reopened and read from at will. To *change* the record that was locked above, this procedure is used:

```
CLOSE n                        Close the desired file.
WAIT FOR 1*                    Wait for record writing permission.
OPEN"filename",n               Open desired file.
INDEX[n] = . . .               Set Index to desired record.
L$ = "0":PRINT%n,L$, . . .     Write the changes to the desired
                               record and unlock it.
CLOSE n                        Close the file.
WAIT CLEAR 1*                  Permit others to lock/write records.
```

WAIT FOR Time Limits

The amount of time a 'WAIT FOR n' command will wait can be limited to zero to fifty-nine seconds with the following command:

POKE 19632,s
where s is the number of seconds, 0–59.

If zero seconds is specified, one check of the semaphore will be made followed by an immediate return to the user program. If

sixty of more seconds are specified the program will remain suspended indefinitely or until the semaphore is available.

If a limited wait (0-59 seconds) has been specified the program must check the contents of location 19633 after executing the WAIT FOR to determine if the semaphore has been locked on his behalf. This is done with a command like:

IF PEEK (19633) GOTO . . . :REM GOTO IF HAVE RESOURCE

If the PEEK yields zero a timeout occurred because the resource remained locked by another user. If the PEEK location is non-zero the resource has been locked for the caller.

This feature can be used in a program to limit the wait for a shared resource so that the computer operator can then be asked whether or not he wishes to continue waiting.

This feature is also useful when more than one shared resource must be acquired by a program. Since there is a possibility of a deadlock situation arising under such circumstances each program that must acquire multiple shared resources should use either ordered locking or a dedicated hypothetical resource which is locked whenever acquiring resources.

The ordered locking method is facilitated by the use of WAIT FOR time limits. For example, the following commands could be used to lock a set of N resources assuming that the resource numbers are in the array RN(N) and are in order:

```
  FOR L = 1 TO 10               Limit to 10 attempts
  POKE 19632,time limit<60      Specify time to wait for
                                each resource
  FOR I = 1 TO N
  WAIT FOR RN(I)                Acquire each resource in turn
  IF PEEK(19633) = 0 GOTO x     If unsuccessful GOTO
  NEXT I                        Else continue acquiring
  REM HAVE ALL RESOURCES
      .
      .
      .
x IF I = 1 GOTO y               Failed to get all resources
  FOR J = 1 TO I
  WAIT CLEAR RN(J)              Release those that were ac-
                                quired
  NEXT J
y NEXT L                        Continue trying for 10 at-
                                tempts
      .
      .
      .
```

The particular order in which the resource numbers are acquired does not matter as long as all users utilize the same order.

Automatic Semaphore Clearing

All semaphores locked by a timesharing or Level 2 user are automatically unlocked whenever the direct (command) mode of BASIC is entered and when a CLEAR or RUN command is executed by a BASIC program. There is a limit of 16 semaphores that can be locked at one time. If this limit is exceeded an OM ERROR results.

If a user clears a semaphore he did not currently have locked an FC ERROR results.

Automatic semaphore clearing can be disabled and re-enabled by selected commands.

Monitoring Network Traffic

Network Monitor Messages

The Network Monitor program occupies one memory partition in each node (hard disk based Level 3 computer). A terminal need not be connected to this partition. If a terminal is connected to this partition it will display a record of all node message traffic.

For each message handled by the Network Monitor a single line message is output. The format and examples of these messages are shown here followed by an explanation of each field.

User Type	User Number	Source Node	Msg. Type	Dest. Node	Disk Adr./ Sem. No.	Time
TS	6	K	RQ	M	25088	5:17
L2	0	L	WFR	N	205	6:10

User Type:	TS—Timesharing User L2—Level 2 User
User Number:	0-15
Source Node:	Node to which the user is connected
Message Type:	RQ—Request to read from a block (3584 bytes) RD—Read a block from disk RC—Receipt of a block SE—Sending a block to be written WR—Writing a block to disk CF—Confirming a block written to disk WFR—WAIT FOR request WFC—WAIT FOR confirm WCR—WAIT CLEAR request WCC—WAIT CLEAR confirm
Destination Node:	Node from/to which the user is transferring a block or where the semaphore is located.
Disk Address:	Disk address from/to which block is to be transferred.
Semaphore No.:	For WAIT messages, the semaphore number.
Time:	System time when message was processed, minutes:seconds.

If an error occurs the above message is preceded by the following:

***ERROR Number Port . . .

where: Number is the error number.

Port is the number of the port to/from which the transfer was occurring when the error was detected.

Disabling Monitor Messages

Printing messages does take a small amount of time that could be used by the Network Monitor program for processing additional network traffic. Consequently, under heavy loading conditions all but error messages or even all messages may be eliminated. (All unrecoverable errors are ultimately reported at the message initiator's terminal, anyway.)

Network Hardware Configuration

The hardware requirements for a network system will vary with the number of nodes, Level 2 users and timesharing users required. A minimum configuration would consist of one network node with one Level 2 system connected to it. Expansion of the system would consist of the addition of nodes and Level 2 systems.

Figure 1 shows the maximum configuration of an OS-65U Network System. Any contiguous subset of this configuration is also a valid network configuration if it includes at least one Level 3 computer.

Each terminal shown Figure 1 can be connected to its associated computer by up to fifty feet of cable, or if modems are used, an unlimited distance via telephone lines. The high speed links are limited to a maximum length of 10,000 feet.

The port number into which a Level 2 user is wired determines his user number. However, there is no particular priority associated with any user number—each is given equal opportunity to access the network. Consequently, the assignment of port numbers is strictly arbitrary. Also, there is no software reconfiguration required dependent upon the port numbers selected since all network messages are *initiated by a user* and the Network Monitor merely acts on messages it receives and sends responses back to the initiator.

A network code consists of a Level 3 system as described in the last journal with an additional serial interface board. This would be a CA-10-X with one to sixteen serial ports populated depending upon the number of Level 2 users connected to the node. Port 15 of this board is dedicated for communications between network nodes. Ports zero through 14 are for communicating with Level 2 systems. Thus, if three Level 2 users were connected to the node, a total of four serial ports would be needed. Each port supports a high speed serial communications link which can run as fast as 500k bits/sec. Other than this board, a network node is identical to a Level 3 system. All network nodes must have a minimum of two memory partitions as one partition is used for network support. As an example, if two timesharing users were connected to the node, a total of three memory partitions would be required. It should be noted, however, that regardless of the number of Level 2 systems connected to the node, only one partition is required for network support.

A Level 2 computer consists of a Challenger II or Challenger III computer with terminal. The memory requirements are 56 kilobytes of memory. The base 48 kilobytes of memory would consist of the memory boards discussed in the previous article. The additional 8 kilobytes of memory are on the 555 board which must be installed in the Level 2 system. The 555 board must be used as it also has the serial port on it for the high speed link to the network node. As mentioned in the previous journal, the 555 board can also be used to interface parallel printers and serial devices to the Level 2 computer. With the exception of a new C2-NET computer which Ohio Scientific is introducing, a Level 2 computer requires a floppy disk drive for booting up the network software it requires. A C2-NET computer has software in ROM that allows it to boot up through the high speed serial link from the network node.

As can be seen, expansion of a Level 3 system to a network node requires a minimal amount of additional hardware. With the addition of Level 2 computers, which are standard Ohio Scientific computers with the 555 board added, a powerful distributed processing system can be implemented.

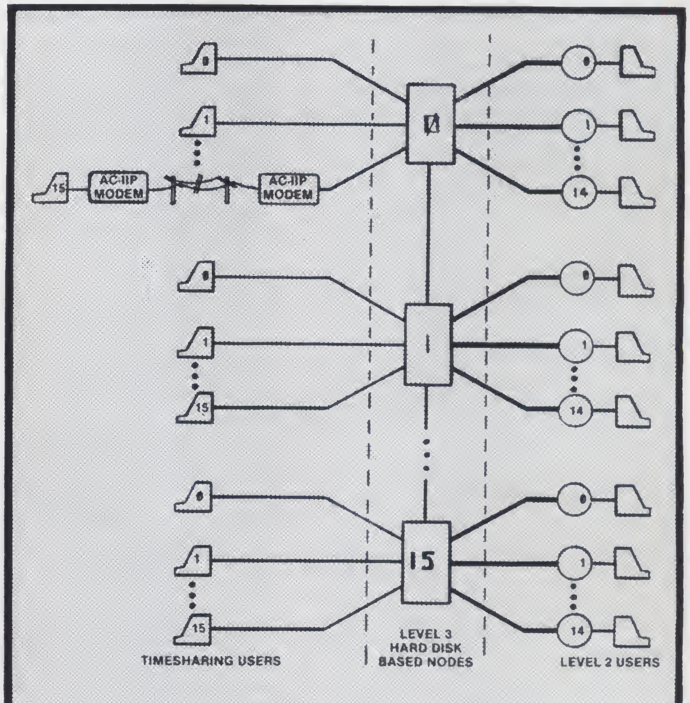


Figure I

KEY:

SERIAL VIDEO OR INPUT/OUTPUT PRINTER TERMINAL

HARD DISK BASED CHALLENGER III – LEVEL 3 COMPUTER

FLOPPY DISK BASED CHALLENGER II OR III COMPUTER OR C2-NET COMPUTER WITHOUT FLOPPY DISK.

LOW SPEED RS-232C LINK (110-19200 BAUD) – MAY INCLUDE MODEMS

HIGH SPEED LINK (UP TO 500K BITS/SECOND)

PET-POURRI

(from page 9)

lowing lines with a lower number can never be found by a BASIC search. This means you probably cannot GOTO or GOSUB to any of the lines following the modified lines with large numbers.

There is also a way to create BASIC lines longer than 80 characters that cannot be entered from the keyboard as a single line. Enter the text as separate lines, then poke the end-of-line flag of the first line, and poke the link and line number of the second line to different values making them part of the new expanded BASIC line of text. The expanded line will list and run correctly, but cannot be edited since it will be truncated if reentered.

This suggests a possible utility program to shrink a BASIC program by stripping all unnecessary line numbers, links and end-of-line flags. It would create long lines of text, and only lines used as targets of GOTOs or GOSUBs would remain. However, the utility would have to handle IF . . . THEN statements since lines following them could not follow and operate correctly. This would also help protect a program being distributed. Although it could not be edited, it could still be listed and saved.

I've also been using POKE within a BASIC program to create computed GOTOs or to have a program permanently modify its logic flow each time it runs. Another idea is to use POKEs during a program to store data within program DATA statements. This provides an alternative to using tape data files for small quantities of data, and the data is readily available through READ statements as often as needed. This technique of saving data within a BASIC program can be applied to most machines.

Saving Tapes

I once had a finished program on tape that would not run after loading, even though there were no load errors. Examination of the tape revealed several areas of the program that appeared garbled. With the information gained through my experimenting, I was able to reconstruct the damaged areas of the program and salvage the program. I now keep backup copies of every program. Using C60 tapes for backups, I can fit 40 to 50 programs on each tape.

More Joystick Information

Since writing my January column I've received more information on the joystick interfaces available from Creative Software, PO Box 4030, Mountain View CA 94040. They have a dual joystick interface for $45 (plus $1.50 shipping) that allows you to connect two Atari joysticks with no modification to the PET. Each joystick can sense the eight compass directions—N, S, E, W, NE, NW, SE, SW—in addition to the red firing button. The sticks can be sensed independently, making them ideal for interactive two-player games.

For more advanced games, they have a single Fairchild joystick interface that costs $35 (plus $2.50 shipping). The Fairchild joystick features eight compass directions and pull-up, push-down, twist-right and twist-left actions. These actions can all be sensed independently.

Each interface comes with a separate power supply, two sample game programs and complete programming instructions. Actual joysticks are *not* included and must be purchased separately at $15 each (plus $1.50 shipping). Both interfaces will work with any model PET, but you should indicate which model you have or use.

Word Pro III

At the Consumer Electronics Show in Las Vegas, Commodore Business Machines announced a word-processing software package for the 32K CBM/PET which offers capabilities formerly available only on more expensive dedicated word-processing systems. Designated Word Pro III, the new software features global functions, instant editing and full documentation retention for up to 170 pages online. Word Pro III can edit an entire diskette of 170K bytes.

The Commodore software system simplifies text entry and editing with a complete range of screen-positioning commands and over 50 control functions, including center titles, indent paragraphs, set tabs and hyphenate words. Real-time screen editing provides such functions as delete, insert, erase, move, search and replace. Standard business form letters can be merged for printing automatically with separate client files such as account names and balances due. A status line at the top of the screen always indicates functions in progress by displaying the text line and column location.

For hard-copy output, formatting features that can be specified include line spacing, left or right margins, justification and multiple printed copies. Word Pro III operates on Commodore's CBM 2022 and CBM 2023 matrix printers, although it is also compatible with NEC, Diablo and Qume printers for letter-quality output.

This new software package complements the previous word-processing packages for the 8K PET and 16K CBM/PET, designated Word Pro I and II. Each package provides the maximum features and functions possible for the amount of memory available. Word Pro III will only operate on a 32K CBM/PET with dual disks. I hope to have more detailed information in the near future.

Miscellaneous Information

Compute magazine has taken over *6502 User Notes*, as well as *PET User Notes* and the *PET Gazette*. *Compute* also has started a cassette tape exchange that is essentially the same as the one operated by Gene Beals and *PET User Notes*. Gene will operate the new exchange; check with *Compute* for complete details.

Excel Company, 618 Grand Avenue, Oakland CA 94610, recently sent me information and pricing on the TX-80 dot-matrix printer for the PET. Excel is one of several companies carrying this new 80-column impact, 150 cps printer. It has double width characters and all the PET graphics as standard features.

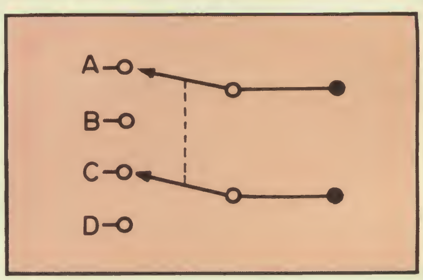
Excel's price is low, starting at $560 with friction feed and $585 with tractor feed. However, the PET interface is an additional $60 and the interface cable is another $25. Because of the difference in the character set in the new ROM set, new PETs require an additional board along with the IEEE interface board at a cost of $25. I have seen the same printer advertised elsewhere for as high as $899.

Please address all correspondence directly to Robert W. Baker, 15 Windsor Dr., Atco NJ 08004.

CORRECTIONS

John Mein wrote to correct some errors in his article, "Color TV Display" from the February issue. There are three errors on the schematic: on IC11, pins 14, 13, 12 and 11 (D1 through D4), should be connected to IC8, pins 23, 22, 21 and 20 (D0 through D3), respectively. The spare half of IC24 (74LS139) should show pin 15 connected to carrier 9, pin 4. The schematic correctly shows IC2 to be a 74LS241, but the parts list incorrectly calls it a 74LS244.

Gregory Yob sent us two additions to his article, "The Comprint Printer" in the March issue. 1. Comprint offers a Centronics-compatible interface. 2. The accompanying figure shows the DPDT switch installed for use with his Imsai. The DPDT switch replaces the Comprint AB and CD jumpers to permit rapid switching from the IEEE 488 to parallel modes. Since the Comprint senses these on power-up, the printer must be turned off and on after you change the switch setting. If you use a "narrow" strobe, an additional switch must be added for the K jumper.



Edited by Dennis Brisson

NEW PRODUCTS

Diskette Protectors

Reviewed by Kevin Cohan, ISI staff.

Two new products from INMAC and Tri-Star Corp. allow the computerist to protect diskettes from the infamous "crunching" effect, caused by misalignment of the disk drive clutch and the center of the floppy disk. Once this damage has happened, the reliability of the disk may be in question.

Installation of a ring of reinforcing material at the center of the diskette will prevent this from happening. Both of these kits allow you to install such a ring.

The Fortifier is available from INMAC (International Minicomputer Accessories Corp.), a nationwide supplier of computer goods. The kit consists of an installation tool and a supply of reinforcing rings for both 8 and 5¼ inch floppies. The tool is a two-part device that presses the ring, made of white vinyl, onto the diskette center. After installation, the disk is protected from "crunching" by the resilience of the vinyl. The tool with 20 rings is $27.

Functionally identical to the Fortifier, the Mini Floppy Saver from Tri-Star Corp. installs center rings on mini-floppies only. It also uses Mylar, instead of vinyl, rings, which seemed sturdier than the ones from INMAC. The installation is identical. The Mini Floppy Saver, with 25 rings, is $14.95.

The installation tool from INMAC seated the rings from both companies more easily; with a little work the Tri-Star tool also provided good results.

INMAC, 2465 Augustine Drive, Santa Clara CA 95051; Reader Service number 484. Tri-Star Corp., PO Box 1727, Grand Junction CO 81502; Reader Service number 485.

The diskette Fortifier from INMAC.

Artec Computer

The Centurion is a new 8-bit small-business microcomputer capable of processing data at 7 MHz. It is built around Intel's 8085A-2 microprocessor, which has a processing speed of 5 MHz, but system speed is much faster because a floating point math chip is used for numerical calculations.

The Centurion features 16K of internal PROM, 64K of RAM, a floppy disk controller, CP/M operating system, built on Artec's shielded motherboard. It operates with a CRT terminal and up to four single-sided, double-density, 8-inch floppy disk drives. It is compatible with any printer having an RS-232 interface.

The Centurion is available in three configurations that differ in packaging and peripheral options: Centurion I with a Hazeltine 1500 CRT terminal is $10,825; Centurion II is $9500; and Centurion III is $8025. An additional megabyte of disk storage capacity ($2500) with two 8-inch drives and a power supply in a separate enclosure can be purchased for the Centurion I and II systems.

Artec Electronics, 605 Old County Road, San Carlos CA 94070. Reader Service number 475.

Artec's new Centurion Microcomputer System.

Earned Income Payroll

This small-business payroll software package handles full-measure payroll activities for firms with up to 80 employees. It contains the new Earned Income Credit provisions which will be required in July 1980, plus federal and state tax tables for any state requiring it. Special city, county or district tax deductions are preprogrammed at no extra charge.

The program requires the CP/M disk operating system, C-BASIC or C-BASIC2, and is available on 8 inch single or double density and Micropolis Mod II 5¼ inch disks. Price is $595.

California Business Computers Corp., 825 W. Hamilton Ave., Campbell CA 95008. Reader Service number 486.

Lowercase for Your Apple

Now you can gain full advantage of uppercase and lowercase on your Apple II with the Keyboard Expandor, from C & H Micro, PO Box 249, Clifton Park NY 12065. This hardware/software modification transforms your Apple II into a complete uppercase/lowercase system.

The software is transparent to the user and compatible with DOS and requires ¼ K memory. Cap and shift locks are included; all Apple characters and Monitor editing functions are maintained. Uppercase/lowercase can be used in Text files, in Print and Rem statements within BASIC programs, in DOS file names and in Immediate mode. Software is available on disk.

A simple one-wire modification with one solder point gives you the use of your shift keys. Price is $20. Reader Service number 483.

OSI High-Speed Sort Utility

For fast operation with Ohio Scientific floppy- and hard-disk systems, consider an enhanced version of BPSort, a high-speed, machine-code sort/merge utility. Twenty thousand bytes can be sorted in ten seconds.

Files can be up to an entire hard or floppy disk in length. BPSort handles fixed-length records. Five keys—alpha or numeric characters—can be specified for ascending and/or descending sequence. Sort parameters are established using an easy-to-use interactive BASIC program.

BPSort, written in assembly language, is OS-DMS compatible and is supplied as part of the BPS interactive data management system. Price is $124. Owners of the previous "V" version may return their original diskette for an update for $25.

BPS, 322 West 57th Street, New York NY 10019. Reader Service number 487.

George Morrow's DISCUS M26 system.

26M Hard Disk Memory

DISCUS M26 is a 26-megabyte hard disk memory for S-100 microcomputer systems. It features the Shugart 4008 14-inch hard disk drive, a sealed media unit utilizing the latest Winchester floating head technology. The drive comes with a metal cabinet, power supply and cables and can be used either as a table-top or rack-mount unit.

The system includes a single-board controller with on-board intelligence to supervise all data transfers. The controller generates interrupts at the completion of each command to increase system throughput. Communication with the CPU is via four I/O ports, command/status, data and control. A 512-byte sector buffer is on-board, and each sector can be individually write-protected for data base security.

The total capacity of the DISCUS M26 is 29 megabytes, with 26 megabytes of usable memory available after formatting. The system can be expanded with up to three Shugart hard disk drives to a maximum of 104 usable megabytes. (Up to three additional drives can be accommodated for a total formatted capacity of 104 megabytes.) DISCUS M26 operates with CP/M 2.0. Price is $4995; additional disk drives are $4455 apiece.

Morrow Designs, Inc./Thinker Toys, 5221 Central Ave., Richmond CA 94804. Reader Service number 478.

Whale of a Product

Melville Technologies is making waves with their announcement of a breakthrough in magnetic media data storage. The product, known as the Moebius disk, doubles the capacity of any standard 8- or 5.25-inch disk drive using conventional single- or double-density recording methods.

While details of the manufacturing process are still being held under wraps, Melville has revealed that the key to the new medium is a proprietary white oxide formulation applied to the disk during a patented convoluting process.

One advantage claimed for the Moebius disk is ease of duplication. Disks may be duplicated off-drive using only a pair of scissors and adhesive tape.

Current technology precludes use of the disk with dual-sided drives, but Melville states that it will have this problem licked as soon as its factory is rebuilt.

Availability: Sooner or later.
Delivery: Twice as long as you expect.
Warranty: About three lines.
Price: To be announced.
Compatibility: Does it really matter?

For further information, call M. E. Ishmael at (123)686-7923, or write Melville Technologies, Inc., 707 Pequode St., New Bedford MA 98765. Reader Service number 501.

Peripheral Control Unit

The Busy Box from The Micromint, Inc., 917 Midway, Woodmere NY 11598, facilitates wireless remote control of ac-operated lights and appliances throughout the home or office. It converts program commands into an ultrasonic message, which is transmitted to the BSR X-10 (Sears) Home Control System. It is signal compatible with most computers and includes complete on-board port addressing.

To turn on a light, you just enter the time and function, e.g., 0730, lamp 2, on. Applications include automatic lighting, energy management and alarm systems.

The Busy Box comes with enclosure, cable, appropriate adapter and complete documentation. Installation is a simple matter of plugging in one connector. It is available for TRS-80 ($104.95), Apple II ($109.95) and S-100 ($114.95). Reader Service number 479.

Word Processor for 6502s

WP-6502 handles word processing for 6502s. Besides screen editing and global editing (with echo-checking and 200+ character insertion), the program's features include:

- AP style—every page starts with a new paragraph.
- Intelligent tabbing—allows tabbing to fixed positions rather than to just the adjacent one.
- Text block files—allows insertion of up to 100 blocks of text anywhere in the text.
- One version supports disk and/or tape.

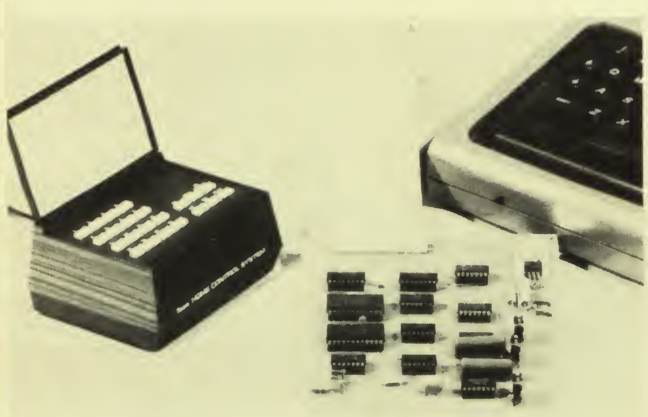
Available in tape or disk, WP-6502 allows generation of OSI C1, C2 and 4P systems. Price is $75.

Dwo Quong Fok Lok Sow, 371 Broome St., New York NY 10013. Reader Service number 492.

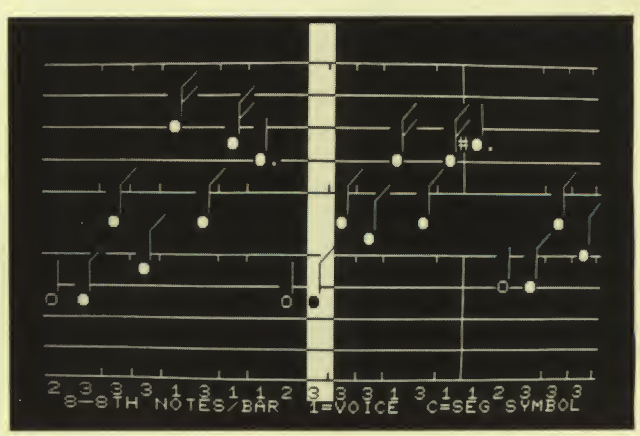
PET Music System

Now PET users can create and play musical compositions of up to four parts with the KL-4M DAC Board and the Visible Music Monitor. The KL-4M Board includes an 8-bit digital to analog converter, a low pass filter to eliminate high-frequency computer-generated hiss and an on-board audio amplifier. An RCA-type jack is also included for quick attachment of your speaker. Amplification of the 6522-CB2-generated single note sound is incorporated as well, so that no additional hardware (other than a speaker) is required. Connection is made via the PET parallel and cassette ports, which are extended with duplicate connectors (with keyways) so I/O capabilities are not reduced in any way.

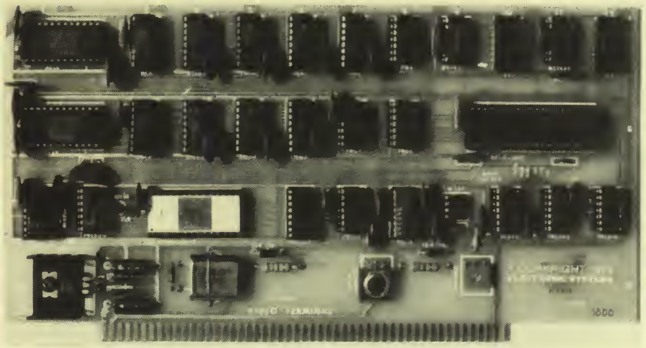
The Visible Music Monitor supports four-



Micromint's Busy Box controller.



PET screen display with the KL-4M and VMM.



Electronic Systems' video board.

Mainframe system from CMC.

part harmony systems such as the KL-4M. VMM is written in 6502 machine language to display the musical staff and notes for all four voices on the PET screen. It provides an easy way to enter four-part music from the keyboard, as well as complete edit capability (including note insertion and deletion). Other features include "record changer" mode to load successive songs without intervention, user definable keyboard, complete tempo flexibility, transpose capability and waveform modification capability. The KL-4M and VMM together cost $59.90, or separately, $34.90 and $29.90, respectively.

A B Computers, 115 E. Stump Rd., Montgomeryville PA 18936. Reader Service number 482.

B/W Monitor

The Video 100-80 is a new 12 inch black and white monitor from Leedex Corp., 2300 E. Higgins Rd., Elk Grove Village IL 60007. Built and styled for industrial use, it includes a rugged metal cabinet. The removable face plate provides mounting space for a mini-floppy disk. There is also a space inside the cabinet for an 11 × 14 inch PC board for custom-designed controller electronics.

The 90 degree deflection picture tube allows an 80 character by 24 line display, and the 12 MHz band width provides well-defined characters. Vertical and horizontal hold, contrast, brightness and power are front-mounted for easy access. The cabinet comes in an off-white color with a black face plate. It is plug-in compatible with Apple, Atari, Radio Shack, OSI, Microterm and Exidy. Reader Service number 480.

S-100 Video Terminal

Electronic Systems, PO Box 21638, San Jose CA 95151, announces a video terminal board for S-100 bus microcomputers that features a 16 line × 64 column display of uppercase/lowercase characters in 5 × 7 dot matrix form, full RS-232 compatibility and a jumper-selectable baud rate generator. The processor chip used is SFF96364 by Neculonic. Control characters include carriage return, line feed, cursors right, left, up and down, nondestructive cursor, clear screen and home. It displays white characters on a black background or vice versa.

By adding a keyboard, video monitor and power supply, you will have a complete standalone terminal. It requires ± 16 V dc at 100 mA and 8 V dc at 1 A. Price for the kit is $199.95. Reader Service number 481.

Microcomputer Mainframe

The Model 2018 Microcomputer Mainframe System consists of an 18-slot S-100 bus motherboard housed in a heavy-duty precision-formed cabinet that is convertible to either a desk-top (Model 2018D) or rack-mounted (Model 2018R) unit. A double-bitted security key lock and a large power-on indicator light built into the reset switch are standard on both models.

The fully shielded motherboard provides interconnections for up to 18 printed circuit cards using the standard S-100 bus format. A jumper system provides active or inactive termination on the various signal lines.

A constant voltage transformer (CVT) selects input voltages of 120 or 230 V ac at 50 or 60 cps. The input power is protected with a double pole circuit breaker on the rear of the cabinet and operationally by the key switch on the front panel. Secondary voltages, which are fully fused, are rated + 8 V dc at 20 Amps and ± 16 V dc at 3.5 Amps. The cooling system funnel design centers the muffin fan so that it exhausts the hot air at the rear of the chassis while drawing cool air through side ducts to the front of the cabinet for maximum cooling efficiency.

CMC Marketing Corp., 10611 Harwin Drive, Suite 406, Houston TX 77036. Reader Service number 477.

Universal Data Entry System

ENTRY increases operator efficiency and accuracy in data entry. It is made up of two programs: UDEGEN and ENTRY. The interactive UDEGEN program generates the custom key-to-disk modules, which are stored as data files to be used with the ENTRY program for actual data entry. It can also be used to revise a previously defined data entry module. It typically requires less than 5 minutes to define a custom module starting cold at the terminal.

Leedex's Video 100-80.

The sequence of entering the data, the CRT headings and labels and the number of records displayed are defined in UDEGEN. Validation procedures such as check digits, tabled value tests, range tests, batch totals and record counts are provided. Field items can be duplicated or incremented to eliminate repetitive entries. User-defined fixed and variable length disk records are supported and easily implemented. You need an 8080 or Z-80 mainframe with 48K of memory, floppy or hard disk, CRT and optional printer for ENTRY, which operates on CP/M with Microsoft MBASIC or Mits/Pertec Disk Extended BASIC.

The Software Store, 706 Chippewa Square, Marquette MI 49855. Reader Service number 491.

Daisy Wheel Printer

The HY-Q 1000 is an intelligent printer for personal computers in business applications. With its five built-in microprocessors, the HY-Q 1000, a low-cost, letter-quality daisy wheel printer, eliminates the need for complex personal computer software. Now microcomputer owners can plug any computer into an HY-Q 1000, which will automatically convert simple codes into instructions for commonly used text-formatting functions.

Other advanced features include Quadra-Pitch (10, 12 or 15 cpi or proportional spacing); up to 198 characters per line; 100 printable characters in five languages (English, Italian, Spanish, French and German, available without changing the daisy wheel); a choice of 21 different typestyles in five different colors; and reverse printing—white characters on a black background. It can also function as a versatile electronic typewriter. Price is $2495.

XYMEC, 17791 Skypark Circle H, Irvine CA 92714. Reader Service number 476.

Morloc's Tower

Did you ever fantasize you had to match wits against an evil wizard to rescue an entire city? This is just what happens in the new fantasy game, Morloc's Tower, from Automated Simulations, PO Box 4232, Mountain View CA 94040. You must hunt through a maze of 30 rooms—all displayed on the screen—in search of the elusive Morloc before he destroys the city of Hagedorn.

Morloc's Tower combines a challenging puzzle to solve with graphics and 18 real-time command options. Dozens of frightening monsters of different shapes and sizes leap from the shadows to assault the player. Three kinds of rings, a magic sword, two amulets and a half dozen other treasures are hidden within the six-floor tower to aid, or hinder, the adventurer.

The competitive scoring system keeps the game challenging and exciting even after many of the tower's mysteries have been revealed. Three levels of play let the user adjust to the difficulty of the game, while the Book of Lore not only explains the rules, but also offers helpful hints on solving the puzzles.

Morloc's Tower is designed for the Commodore PET (with at least 20K), the Radio Shack TRS-80 (Level II, 16K) and the Apple II (32K with Applesoft in ROM). Price is $14.95. Reader Service number 488.

COMPUTER CLINIC

I would appreciate any and all information on schools, institutions, companies and individuals involved in teaching and/or building single-chip microcomputers used specifically for digital controls.

Don Wilson
9055 S. Luella
Chicago IL 60617

I recently purchased Appleforth and am enjoying it. However, because of its unusual structure and nature, it is difficult to learn enough to fully exploit the power of FORTH. I want to get in touch with readers who are interested in a newsletter devoted to exchanging information on the use and application of FORTH.

H. John Clements
9010 Tobias #258
Panorama City CA 91402

I'm head of the computer programming department at the local high school. We have several 16K PETs and a terminal hooked up for time-sharing. We want to hook up the PETs to the Decwriter II so we can get hard copy. We need information on how to connect the PET to the terminal.

Dale Freeland
Paw Paw High School
Paw Paw MI 49079

I work for the city of Quincy MA as superintendent of fire alarms and have written some simple programs to keep our files up to date and to store the normal "garbage" required by all government units from village to federal levels. However, I can use some help. My fire alarm files, boxes, billing and circuit listing are in good shape, but the NFPA has designed an "incident reporting system" for all fire departments. The fire reports are designed for entry into a computer. It is assumed that all fire departments have access to hardware in the mega-buck range and the "incident" programs are so designed. Great, if the "big" equipment is at hand. I guess that many cities and towns within the US of A would like to be able to purchase smaller computers and have an in-house system (and control) to have the fire-department records at their fingertips. Programming experience to solve this problem will take time. We would be interested in making contact with some sharp fire-department "type" out there who is working on this problem or may even have a working program.

John E. Schmock
Apt. 55, 10 Mediterranean Dr.
Weymouth MA 02188

I built an ASCII-encoded keyboard from scratch in hopes of making a computer terminal; however, I cannot find an 80 × 24 TVT in kit form or otherwise, short of buying a $2000 professional computer terminal. If I can't locate an 80 × 24 TVT that costs approximately $200 or less, I want to know how, if possible, a more common 64 × 16 TVT can be modified to print out 80 × 24. I do not have a microprocessor, and I intend to modify an old TV to use as a video monitor.

Stuart Weiner
65-23 Dieterle Cresc.
Rego Park NY 11374

I met a man who is retired and on a fixed income. He is afraid to spend any of his savings on a computer because he may need the money for silly things such as food. Since he lives in another city, I can't help by lending him the use of my TRS-80. I wonder if any computer fanatics in the Portsmouth Ohio area can give him a hand. His address is: Earl Keevil, 1604 4th, Portsmouth OH 45662.

Adam Shackleford
48 Elm
Canal Winchester OH 43110

I'm looking for a company that supplies the solenoids to make an IBM Selectric typewriter operate as a printer.

Donald McKague
PO Box 227
Teeswater Ontario
Canada

✓Reader Service—see page 257

LETTERS TO THE EDITOR

9368 Et Al

It's too bad that Robert Cotter had such difficulty sourcing the Fairchild 9368 hex decoder-drivers for his Elf expansion. 9368s are available at reasonable prices from hobbyist-oriented mail-order distributors, notably: Active Electronic Sales, Framingham MA, Jade Computer Products, Hawthorne CA, Advanced Computer Products, Irvine CA, Quest Electronics, Santa Clara CA.

Your readers may also like to know about three other integrated circuits that do what the 9368 does: latch a 4-bit binary input, convert it to a 7-segment representation of a hexadecimal number and provide enough output current to directly drive a common-cathode LED display. Because these other chips have CMOS inputs, they require no input buffering and don't load the computer's bus at all. And because they have bipolar outputs, they can directly drive the LED display. They also consume virtually no current on their own, unlike the 9368's typical supply current drain of 45 mA.

Two of the ICs are made by Mitel, a relatively obscure Canadian CMOS manufacturer. Mitel's MD4368 is a pin-for-pin and function-for-function replacement for the 9368. Mitel also makes an MD4311, which is similar to the well-known 4511 offered by Motorola, RCA and many others—but the 4311 is a hexadecimal decoder. The only difference between the MD4368 and the MD4311 is that one has a ripple-blanking output, and the other has a lamp-test input. Both chips are available from Ancrona Electronics, Culver City CA.

The third entry, Motorola's MC14495, has not yet appeared in the hobbyist suppliers' catalogs, but is undoubtedly available from Motorola's usual distributors. This chip lacks the blanking inputs on the two Mitel ICs, but has an output that goes high for all hex numbers over 9, and another output that goes low only on hex F.

Robert Levine
New York NY

It seemed to me that Russell Steele could have bought at least three excellent monitors for the price he paid for an untried TV set ("Bargain-Basement '80," February 1980, p. 54). I was skeptical at first when I ordered a $45 12-inch used monitor from Selectronics (February 1980, p. 222), but the results were so pleasing I bought five more for students in my electronics classes.

Every monitor was running in no more than five minutes. All gave clean, crisp readouts on the TRS-80 as well as the Elf-44. The units were well packed and all were complete, including filter face plate. The stand was easily removed to permit use on the TRS-80 expansion interface. Most of them have been operating eight hours each day for the past four months. Selectronics sells a handbook and will replace any unsatisfactory part or unit.

From what I have been told, Radio Shack no longer sells the keyboard separately—in fact they want $245 for the monitor because it has to be classed as a replacement part. That prompted my dealing with Selectronics.

My second remark concerns a letter on p. 14 of the February issue by Robert J. Cotter who agonized about the scarcity of 9368 decoder chips. The solution is in the magazine. The 9368 chip has always been available from Quest and is still in their advertisement under MOS/MEMORY at $3.50 (February, p. 213).

I am pleased to see the old Elf getting some boost from a magazine as exalted as *Microcomputing*. I made my original Elf (still working) from the RCA user's manual on 44-pin 4×4 inch boards from Radio Shack years ago.
P.S. Congrats on "80 Micro."

Alan Wallace
Goldsboro NC

From the Source

I noticed two errors in James Downey's "Sample the 6100" article (December 1979 issue, p. 54). First, the schematic on page 55 has a component labeled IM6103; it should be labeled IM6403. Second, the statement "DECUS Society . . . membership is limited to users of Digital Equipment's machines" is misleading. Anyone with a bona fide interest in DECUS may apply for associate membership simply by filling out an application. No CPU serial number is required as in the installation membership application.

Mr. Downey's article is excellent, and I am happy to see the IM6100 mentioned in your magazine as I believe it has considerable unrealized hobby potential.

Dave Kocsis
Supervisor, Software Design
Intersil, Inc.
Cupertino CA

Amen

In "Heath's H19: A Detailed Look" (February 1980 issue, p. 58), Ralph Wynkoop recommends scraping "all the way around the leads of all components . . . with a penknife."

I want to say "amen" to this practice and make a recommendation of my own. After scraping all leads (gently, very gently), I clean everything with isopropyl alcohol. I also wipe the solder I am going to use and the PC board pads on both sides. At one time, I cleaned only the component leads. Then I started cleaning the PC board and discovered that everything soldered faster and used less solder, and that there was less flux contamination of the areas around the soldered connection.

One word of caution: do *not* use rubbing alcohol. It contains emollients such as lanolin and will contaminate everything wiped with it.

William J. Hartweg
Staten Island NY

26 to 20

I want to thank Wayne Green and everyone associated with *Microcomputing* for such an excellent publication. It has been interesting watching the magazine grow to its present state.

I also want to thank Pete Stark for his series on the SWTP computer system. My 6800 system has benefited considerably from his articles.

In part 9 (February 1980) of the series, Pete points out a BASIC bug concerning the use of an MP-S interface with SWTP Disk BASIC Version 3.0. The recommended fix consists of changing a byte of memory in location 1472 from 26 to 20. Upon examining memory, I found a 26 not at 1472 but at 1477. Changing the 26 at this location to 20 allowed my newly interfaced Heath H14 printer to come alive.

Darwin Frerking
Garland TX

AMI VDG Source

I've been receiving dozens of calls and letters concerning the S64807 (see February 1980 issue, p. 148). The S68047 is not pin-compatible with Motorola's 6847; the S68047 is available by mail from Advanced Computer Products, PO Box 17329, Irvine CA 92713, (714) 558-8813.

John C. Mein
Arvada CO

Expandoram Tip

Ron Derynck's article on the S.D. Sales Expandoram in the December 1979 issue of *Microcomputing* was really good. I experienced the same problem, and his fix was the key to correcting it. Here is a tip for anyone who has an Expandoram with 8K chips. Wire the board

(see LETTERS, page 238)

A Few Extraordinary Products for Your 6800/6809 Computer


SS-50 Bus LFD-400™ and LFD-800™ Systems

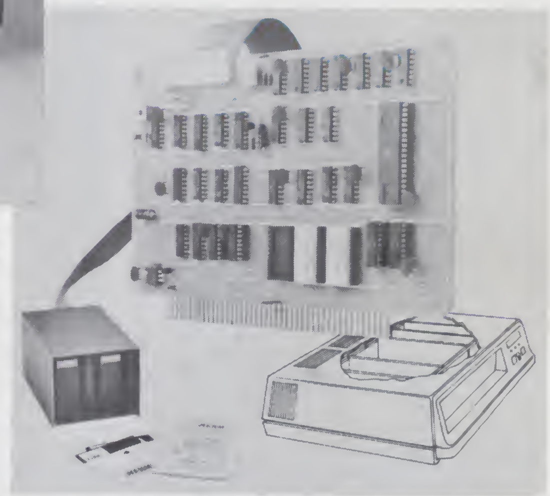
From Percom . . .

13

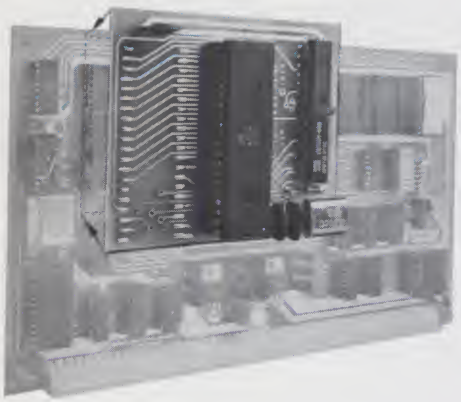
Low Cost Mini-Disk Storage in the Size You Want

Percom mini-disk systems start as low as $599.95, ready to plug in and run. You can't get better quality or a broader selection of disk software from any other microcomputer disk system manufacturer — at any price!

Features: 1-, 2- and 3-drive systems in 40- and 77-track versions store 102K- to 591K-bytes of random access data on-line • controllers include explicit clock/data separation circuit, motor inactivity time-out circuit, buffered control lines and other mature design concepts • ROM DOS included with SS-50 bus version — optional DOSs for EXORciser* bus • extra PROM sockets on-board • EXORciser* bus version has 1K-byte RAM • supported by extended disk operating systems; assemblers and other program development/debugging aids; BASIC, FORTRAN, Pascal and SPL/M languages; and, business application programs.


EXORciser Bus LFD-400EX™ -800EX™ Systems*

14



15 **Upgrade to 6809 Computing Power**

This 6809 upgrade adapter may be used on the SWTP 6800 and most other 6800/6802 MPU cards. Supplied assembled and tested, it costs only $69.95 with user instructions. The original system may be restored by merely unplugging the adapter and a wire-jumpered DIP header, and re-inserting the original components. Also available for your upgrade computer is PSYMON™, the Percom SYstem MONitor for the Percom 6809 single-board computer. PSYMON™ on 2716 ROM costs only $69.95 — PSYMON™ is also available on minidiskette, with source and object files, from the Percom Users Group.

™ trademark of Percom Data Company, Inc.
* trademark of the Motorola Corporation.
Prices and specifications subject to change without notice.

Full Feature Prototyping PC Boards

Percom SS-50 and I/O bus prototyping cards include all of the features needed for easy, straightforward prototyping. Use wire wrap, wiring pencil or solder wiring. **Features**: tin-lead plating over 2 oz. copper wets quickly, solders easily • provision for power regulators and distributed capacitor bypassing • SS-50 bus card accommodates 34- and 50-pin ribbon connectors on top edge, 10-pin Molex connector on side edge — costs only $24.95. • I/O bus card is 1-1/4" higher than SWTP I/O card, accommodates 34-pin ribbon connector and 12-pin Molex connector on top edge — costs only $14.95 • Both card designs accept 14-, 16-, 24- and 40-pin DIP sockets.

The Electric Window™: Instant, Real-Time Video Display Control

This VDC card resides completely in main memory so that control is accomplished instantly by direct MPU access to the on-board 2K character-store memory and the display control registers. Price is only $249.95. **Features**: Programmable CRT controller chip provides extraordinary capability for software control of functions such as number of characters per line, number of lines displayed, highlighting and interlaced or non-interlaced scan • includes ASCII 128-unit character generator which generates 7-dot by 12-dot characters — lower case letters have descenders • provision for optional ROM for special characters/symbols • comprehensive manual includes full listing of WINDEX™, the Electric Window™ driver program — WINDEX™ is also available on minidiskette through the Percom Users Group.

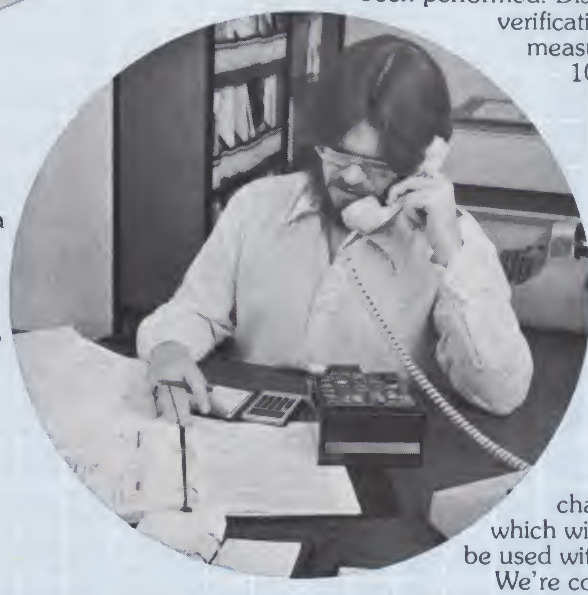
PERCOM

PERCOM DATA COMPANY, INC.
211 N. KIRBY GARLAND, TEXAS 75042
(214) 272-3421

Products are available at Percom dealers nationwide. Call toll-free, 1-800-527-1592, for the address of your nearest dealer, or to order direct.

"From an efficient 1K-byte control system DOS to high level languages such as FORTRAN and Pascal, no other microcomputer disk systems manufacturer provides the range and quality of development and application programs available from Percom."

"Connie is running a 'cats eye' test on a mini-disk drive to check radial track alignment. Drive motor-speed timing and sensor alignment tests have already been performed. Disk formatting and format verification tests are next. These measurements are part of the 100% testing every single unit receives."



"Whether you call about a shipping date or ask a tough technical question, you get a competent courteous answer. Outstanding customer service is a hallmark of Percom."

"Richard's making final changes to a disk controller which will allow Percom drives to be used with yet another computer. We're constantly developing and introducing new products that extend and enhance the value of Percom systems."

"Slipping a circuit board through the eye of a needle would be easier than slipping a cold solder joint past Beverly. These are four-drive LFD-400/800 disk system controllers she's inspecting."

Reader Service—see page 257

Green-Thumb Computing

A computerized cornucopia awaits you if you use this program to plan your garden.

Robert H. Rhoades
Rt. 1, Box 456
Scottsdale AZ 85256

It's time to plan the garden. I decided to use my SWTP system to plan my garden.

After a long think-and-read session, I came up with the following most needed data for planning a garden: planting and yield information; soil information; companion planting; succession planting; best varieties; some text material.

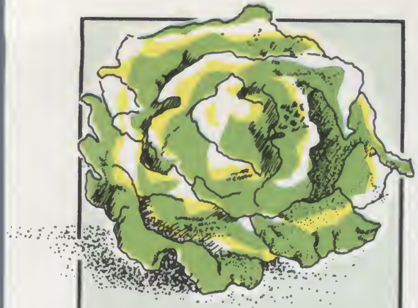
Planting information includes plant spacing in a row, distance between rows, depth of seed, yield per 100 feet. See the program listing and sample run for an explanation of planting and yield information.

Companion planting lists plants that get along with or help the vegetable they are planted next to. For instance, beans planted with corn help promote the growth of both. Onions planted next *to cole* crops (cabbage, broccoli) deter cabbage worms, but they tend to stunt the growth of beans and peas.

Succession plants are vegeta-

Program 1.

```
0020 REM WRITTEN BY R H RHOADES
0050 REM 7-20-78
0060 PRINT "THE GREEN THUMB COMPUTER":PRINT
0070 PRINT "THIS PROGRAM ALLOWS MANY"
0075 PRINT "DIFFERENT FILES ON VEGETABLES."
0085 PRINT :PRINT
0200 DIM A$(10),B$(2),C$(10),D$(10)
0205 DIM E$(10),F$(30)
0300 PRINT "DO YOU WANT TO:"
0310 PRINT "1. CREATE A NEW FILE OR CORRECT"
0320 PRINT "   AN OLD ONE."
0330 PRINT "2. LOOK AT A FILE"
0340 PRINT "ENTER 1 OR 2"
0350 INPUT X
0360 IF X=1 THEN1000
0370 IF X=2 THEN3000
0380 GOTO 300
1000 PRINT "ENTER CODE WHEN ASKED"
1005 PRINT
1006 GOTO 7100
1007 REM - PICK OUT THE CODE OF THE
1008 REM - VEGETABLE WHOSE FILE YOU WANT
1009 REM - TO WORK ON.
1010 INPUT "ENTER VEGETABLE'S CODE",M$
1015 INPUT "VEGETABLE'S NAME",N$
1025 PRINT "PLANTING INFORMATION"
1030 INPUT "DAYS TO GERMINATION  ",A$(1)
1040 INPUT "DAYS TO TRANSPLANTING ",A$(2)
1050 INPUT "DAYS TO HARVEST  ",A$(3)
1060 INPUT "PLANT SPACING IN ROW  ",A$(4)
1070 INPUT "SPACE BETWEEN ROWS  ",A$(5)
1080 INPUT "SEED PLANTING DEPTH  ",A$(6)
1090 INPUT "PLANTING DATES  ",A$(10)
1100 INPUT "PLANTING PER PERSON  ",A$(7)
1110 INPUT "PLANTS PER 100'  ",A$(8)
1120 INPUT "YIELD PER 100'  ",A$(9)
1200 PRINT "SOIL INFORMATION"
1210 INPUT "SOIL TYPE  ",B$(1)
1220 INPUT "PH  ",B$(2)
1295 PRINT :PRINT
1297 PRINT "ENTER 0 IN ANY UNUSED LINES"
1298 PRINT "OF THE NEXT FOUR SECTIONS."
1299 PRINT :PRINT
1300 PRINT "COMPANION PLANTS"
1310 FOR I=1TO10
1320 PRINT I;:INPUTC$(I)
1340 NEXT I
1399 PRINT
1400 PRINT "SUCCESSION PLANTS"
1410 FOR I=1TO10
1420 PRINT I;:INPUTD$(I)
1430 NEXT I
1499 PRINT
1500 PRINT "BEST VARIETIES"
1510 FOR I=1TO10
1520 PRINT I;:INPUTE$(I)
1530 NEXT I
1598 REM - HERE WE CAN ENTER ANY GENERAL
1599 REM - INFO WE WANT TO INCLUDE IN THE FILE
1600 PRINT "GENERAL INFORMATION"
1610 FOR I=1TO30
1620 PRINT I;:INPUTF$(I)
1630 NEXT I
1809 REM - NOW WE WRITE THE FILE ON THE DISK
1810 OPEN #1,M$
1820 SCRATCH #1
1825 WRITE #1,N$
1830 FOR I=1TO10:WRITE#1,A$(I):NEXTI
1860 FOR I=1TO2:WRITE#1,B$(I):NEXTI
1900 FOR I=1TO10:WRITE#1,C$(I):NEXTI
1940 FOR I=1TO10:WRITE#1,D$(I):NEXTI
1970 FOR I=1TO10:WRITE#1,E$(I):NEXTI
1980 FOR I=1TO30:WRITE#1,F$(I):NEXTI
1999 CLOSE #1
2000 INPUT "WRITE ANOTHER FILE  (Y OR NO)",X$
2010 IF X$="Y"THEN1000
2020 GOTO 5020
3000 REM OUTPUT
3010 PRINT "ENTER CODE WHEN ASKED"
3015 GOTO 7100
3017 REM - PICK OUT THE CODE OF THE
3018 REM - VEGETABLE WHOSE FILE YOU WANT
3019 REM - READ OR PRINT-OUT.
3020 INPUT "ENTER VEGETABLE'S CODE",M$
3050 OPEN #1,M$
3060 READ #1,N$
3090 FOR I=1TO10:READ#1,A$(I):NEXTI
3110 FOR I=1TO2:READ#1,B$(I):NEXTI
3140 FOR I=1TO10:READ#1,C$(I):NEXTI
3170 FOR I=1TO10:READ#1,D$(I):NEXTI
3200 FOR I=1TO10:READ#1,E$(I):NEXTI
3210 FOR I=1TO30:READ#1,F$(I):NEXTI
```



```
LEAF LETTUCE

PLANTING INFORMATION
DAYS TO GERMINATION 4 TO 10
DAYS TO TRANSPLANT 20 TO 35
DAYS TO HARVEST 45-60 FROM SEED
PLANT SPACING IN ROW 6-12" FINAL
SPACE BETWEEN ROWS 6" TO 12"
SEED PLANTING DEPTH 1/4" TO 1/2"
PLANTING DATES 9-1 TO 3-1

YIELDS
PLANTING PER PERSON 6-12 FEET
PLANTS PER 100′ 200
YIELD PER 100′ 150-200 PLANTS

SOIL INFORMATION
SOIL TYPE - MOIST AND RICH
PH - 6.0 - 7.0

COMPANION PLANTS
  (A) = ALLIES (HELP)
  (E) = ENEMIES (HARM)
CARROTS
RADISH
STRAWBERRIES-BORDER FOR BERRIES
CUCUMBERS
KOHLRABI (INTERCROP)
ONIONS (A)
GARLIC (A)
BUSH BEANS (A)
POT MARIGOLDS (A)

SUCCESSION PLANTS
SWISS CHARD
BEANS
KALE
CABBAGE FAMILY
CUCUMBERS
TOMATOES
CARROTS
RADISH

BEST VARIETIES
(#) = APPROXIMATE DAYS TO HARVEST
BLACK SEEDED SIMPSON (45) EARLY
SALAD BOWL (48) MIDSEASIN
SLOBOLT (48) MIDSEASON
OAKLEAF (50) MIDSEASON
RUBY (47)
GRAND RAPIDS (45) MIDSEASON
GREEN ICE (45) MIDSEASON

GENERAL INFORMATION
GROWS BEST IN COOLER WEATHER
OR IN SHADY PLACES. PLANT EVERY
2 WKS. MULCH ALONG ROWS AND
BETWEEN PLANTS. SUCCESS DEMANDS
MOISTURE & NUTRIENTS. PLANTS
SHOULD NOT TOUCH FOR BEST GROWTH
USE THINNINGS IN EARLY SALADS.
MAY BE PLANTED A WEEK BEFORE
THE LAST EXPECTED FROST.
```

bles that can be planted after earlier ones have been harvested. This type of planting will extend your production by using space that would otherwise lie fallow. With today's food prices, it makes sense to get as much production as possible. You can then store a portion of the yield to eat later, and at a cheaper price than at the supermarket.

Best varieties are those best suited for your locality and taste. These are best determined by experience—yours and that of others. Most of the data in a file comes from the experience of your style of gardening. For example, I use raised beds, where leaves and other organic matter have formed a rich humus. This allows me to plant at much closer intervals than is normally advised.

Text material, or general information, as it is labeled in the program, allows you to enter tidbits of extra information you might want to include. You are allowed up to 30 32-character lines for this item.

The System

Before explaining the program, I will list the specifications of my system:

1. SWTP 6800 with SWTBUG.
2. 32K of memory.
3. SWTP FLEX DOS and 8K Disk BASIC (3.0).
4. SWTP MF68 dual drive minifloppy. (Only one drive is needed for this program.)
5. SWTP PR-40 printer.

Because the PR-40 types only 40 columns, I set the string variables at 32 characters. If you have a longer line length available on your printer, you can expand this length if you need to.

Slightly over 4K of memory is needed for the program and 2.3K for the file that is generated. 16K plus the 4K needed for FLEX at 7000 to 7FFF would be adequate for the program in its present form.

Now for a step-by-step review of the program. Lines 200–205 set up the dimensions of each of the arrays in memory files.

Writing a File

The 300 series of lines gives you the choice of either (1) creating or correcting a file or (2) looking up a file on the disk. You are asked to enter 1 or 2, which directs the program to either line 1000 or 3000.

Lines 1000 through 2999 are used to write a new file or correct an old one. You can use the TSC Text Editor to edit the file.

Line 1006 directs the program to the 7000 series lines, which contain a list of vegetables and their respective codes. Lines 7300 and 7310 then direct the program back to the proper routine (write or read).

Line 1010 now requests that you enter the alphanumeric code of the vegetable for which you are about to write a file. This code becomes the name of the disk file. Line 1015 requests the name of the vegetable for which you will be entering data.

```
3250 CLOSE #1
3400 PRINT "WANT A PRINT-OUT (Y OR N)";
3410 INPUT Z$
3420  K=1:IFZ$="Y" THEN K=7
3490 PRINT #K:PRINT#K:PRINT#K
3500 PRINT #K,N$
3510 PRINT #K
3520 PRINT #K,"PLANTING INFORMATION"
3530 PRINT #K,"DAYS TO GERMINATION ";A$(1)
3540 PRINT #K,"DAYS TO TRANSPLANT ";A$(2)
3550 PRINT #K,"DAYS TO HARVEST ";A$(3)
3560 PRINT #K,"PLANT SPACING IN ROW ";A$(4)
3570 PRINT #K,"SPACE BETWEEN ROWS ";A$(5)
3580 PRINT #K,"SEED PLANTING DEPTH ";A$(6)
3585 PRINT #K,"PLANTING DATES ";A$(10)
3590 PRINT #K:IFZ$="Y" THEN 3600
3595 INPUT "PRESS 'RETURN' TO GO ON",R$
3600 PRINT #K,"YIELDS"
3610 PRINT #K,"PLANTING PER PERSON ";A$(7)
3620 PRINT #K,"PLANTS PER 100′ ";A$(8)
3630 PRINT #K,"YIELD PER 100′ ";A$(9)
3699 PRINT #K
3700 PRINT #K,"SOIL INFORMATION"
3710 PRINT #K,"SOIL TYPE - ";B$(1)
3720 PRINT #K,"PH - ";B$(2)
3790 PRINT #K:IFZ$="Y"THEN3800
3795 INPUT "PRESS 'RETURN' TO GO ON",R$
3800 PRINT #K,"COMPANION PLANTS"
3802 PRINT #K,TAB(3);"(A) = ALLIES (HELP)"
3803 PRINT #K,TAB(3);"(E) = ENEMIES (HARM)"
3810 FOR I=1TO10
3820 IF C$(I)="0" THEN 3890
3830 PRINT #K,C$(I)
3840 NEXT I
3890 PRINT #K:IF Z$="Y"THEN 3900
3895 INPUT "PRESS 'RETURN' TO GO ON",R$
3899 PRINT #K
3900 PRINT #K,"SUCCESSION PLANTS"
3910 FOR I=1TO10
3920 IF D$(I)="0"THEN3990
3930 PRINT #K,D$(I)
3940 NEXT I
3990 PRINT #K:IFZ$="Y"THEN4000
3995 INPUT "PRESS 'RETURN' TO GO ON",R$
4000 PRINT #K,"BEST VARIETIES"
4002 PRINT #K,"(#) = APPROXIMATE DAYS TO HARVEST"
4010 FOR I=1TO10
4020 IF E$(I)="0"THEN4080
4030 PRINT #K,E$(I)
4040 NEXT I
4080 PRINT #K
4090 IF Z$="Y" THEN 4100
4095 INPUT "PRESS 'RETURN' TO GO ON",R$
4100 PRINT #K,"GENERAL INFORMATION"
4110 FOR I=1 TO 10
4120 IF F$(I)="0" THEN5000
4130 PRINT #K,F$(I)
4140 NEXT I
4145 IF Z$="Y" THEN 4160
4150 INPUT "PRESS 'RETURN' TO GO ON",R$
4160 FOR I=11 TO 20
4170 IF F$(I)= "0" THEN 5000
```

```
4180 PRINT #K,F$(I)
4190 NEXT I
4195 IF Z$="Y" THEN 4210
4200 INPUT "PRESS 'RETURN' TO GO ON",R$
4210 FOR I=21 TO 30
4220 IF F$(I)= "0" THEN 5000
4230 PRINT #K,F$(I)
4240 NEXT I
4250 PRINT #K:PRINT#K
5000 PRINT #K:INPUT"READ ANOTHER FILE  (Y OR N)",Z$
5010 IF Z$="Y"THEN3000
5020 PRINT "END":END
7098 PRINT "CODES AND VEGETABLE NAMES."
7099 PRINT "A IS CODE FOR ASPARAGUS, ETC.":PRINT:PRINT
7100 PRINT "A= ASPARAGUS"
7105 PRINT "B= BUSH BEAN"
7110 PRINT "B1= POLE BEAN";TAB(20);"B2= BEET"
7115 PRINT "B3= BR SPROUTS";TAB(20);"B4= BROCCOLI"
7120 PRINT "C= CABBAGE";TAB(20);"C1= CAULIFLOWER"
7125 PRINT "C2= CARROT";TAB(20);"C3= CELERIAC"
7130 PRINT "C4= CELERY";TAB(20);"C5= CHARD"
7135 PRINT "C6= CORN";TAB(20);"C7= CUKES, HILL"
7136 PRINT "C8= CUKES, TRELLISED"
7140 PRINT "K= KALE";TAB(20);"K1= KOHLRABI"
7145 PRINT "L= LEEKS";TAB(20);"L1= LEAF LETTUCE"
7146 PRINT "L2= BUTTERHEAD LETTUCE"
7147 PRINT "L3= CRISP HEAD OR ICEBERG LETTUCE"
7143 PRINT "L4= COS OR ROMAINE LETTUCE"
7150 PRINT "M= MELONS"
7155 PRINT "O= OKRA";TAB(20);"O1= ONIONS"
7159 INPUT "PRESS 'RETURN' TO GO ON",R$
7160 PRINT "P1= PARSLEY";TAB(20);"P2= PARSNIP"
7165 PRINT "P3= PEAS";TAB(20);"P4= PEPPERS"
7170 PRINT "P5= POTATOES";TAB(20);"P6= PUMPKIN"
7180 PRINT "R= RADISH";TAB(20);"R1= RHUBARB"
7185 PRINT "R2= RUTABAGA"
7190 PRINT "S= SALSIFY";TAB(20);"S1= SPINACH"
7195 PRINT "S2= SUMMER SQUASH";TAB(20);"S3=WINTER SQUASH"
7200 PRINT "S4= SW POTATOE";TAB(20);"S5= STRAWBERRY"
7205 PRINT "T= TOMATOE";TAB(20);"T1= TURNIP"
7300 IF X=1 THEN 1010
7310 IF X=2 THEN 3020
7400 REM - AREA FROM 7500 TO 9999 IS
7401 REM - FOR FUTURE EXPANSION
```

Lines 1030 through 1299 are self-explanatory. Beginning with line 1300, FOR-NEXT loops are used to enter the other data and the general information that you may want to include.

If, while entering data into the file, you find that you do not have enough data to fill in the rest of the prompt lines, type in 0 for each of the remaining input requests. This input tells the output section of the program to go to the next section of the file (note lines 3820, 3920, 4020 and 4120).

Lines 1810 to 1999 write the file onto the disk. Line 2000 asks if you want to write another file, and line 2020 ends the program if you don't.

Read or Print Out a File

The output section of the program begins at line 3000. The program again takes you through the 7000 series of lines to get the vegetable code. After inputting the code in line 3020, the file is read into memory from the disk by lines 3050 through 3250. If a code is entered for which a file has not yet been written, you will get a "NO SUCH FILE" error message. The program will abort and return to BASIC. Type RUN and start over. If you don't remember which files you have on the disk, type CAT and check. Be sure you do this before you type RUN.

Line 3400 then asks if you want a printout. If you answer yes, the machine will output to the printer at port 7. If your printer is on any other port, change the value of the last K in line 3420 to the proper port number. The first value, K = 1, directs the output to the CRT on port 1.

Lines 3490 through 4999 print the information either on the CRT or printer (see the sample run).

There is a delay used in lines 7154, 3595, 3795, 3895 and 3995 to allow you sufficient time to read what is displayed on the screen. You even have enough time to write down the data if you don't have a printer. These pauses have been placed so that my screen (16 lines) is not quite filled. The program continues after you press the RETURN key.

Lines 5000 through 5020 either send you back for another file or end the program.

The development of this program has been a great learning experience for me. I thought I had a fairly adequate knowledge of how to program in BASIC until I wrote this program. Before this, I had written several small programs for my work and altered many ham radio programs to work in SWTP 8K BASIC.

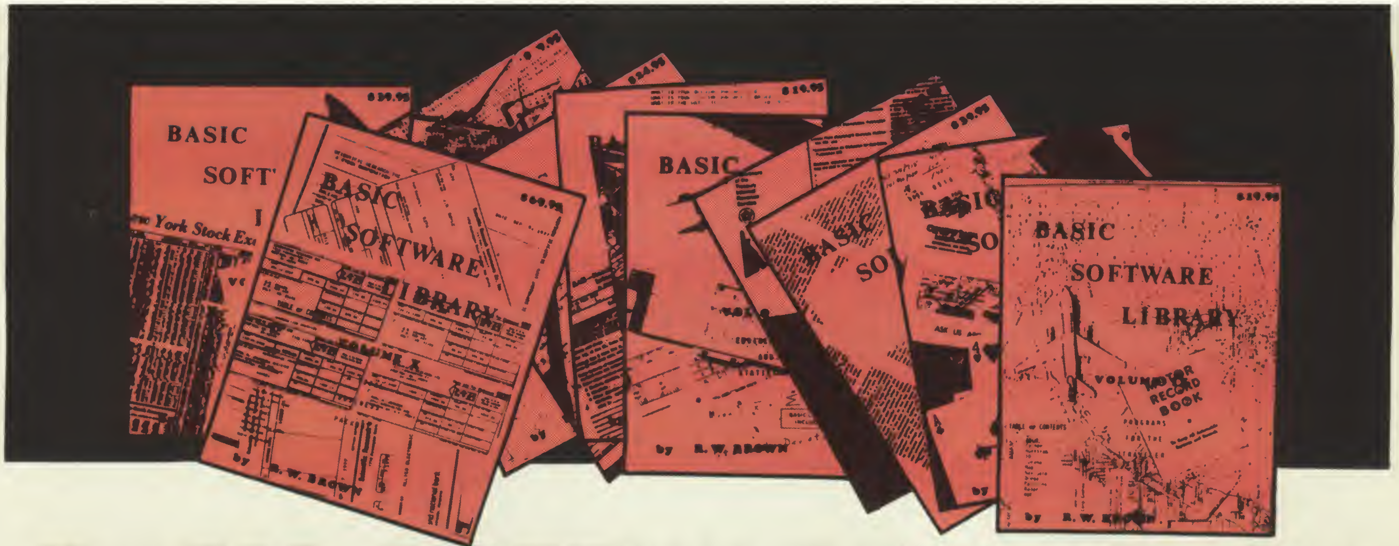
In preparing this article, I used the disk version of TSC's Text Editor and Text Processor for the first time, other than for practice. If you are planning to write an article, get a good editor/processor such as this. TSC now has published 8080 versions.

Afterthoughts

After writing this article, I noticed that, with suitable alterations, the program could be used for other purposes. How about listing hundreds of your favorite recipes? If you reload your own shot shells or rifle and pistol ammunition, you could use this program format. All in all, the general scheme shows a lot of possibilities, and I am sure you will be able to adapt it to your needs.

Those readers with more programming experience than I may see ways to improve the program. Let me know, I am willing to learn.

I made no attempt to adapt this program to saving the files on tape. Tape files would only allow for 26 vegetables (A to Z). In fact, it was because of troubles with tape files that I splurged and bought the minifloppy. Have you ever had to recopy 250 names, addresses and due records three times? The minifloppy has been well worth its cost in time saved and avoidance of problems.

My thanks to Roger Smith of Personal Computer Place for his encouragement and help. A disk copy of this program is available from Personal Computer Place, 1840 West Southern Ave., Mesa AZ 85202, for $10 per disk. ■

BASIC SOFTWARE LIBRARY

NOW ★ 10 ★ Volumes and Growing

IS SPONSORING A

$10000.00 Give Away

And unlike others we are giving a large portion in CASH that you Don't have to spend with us. You are eligible NOW! You may already be entered. More details in future advertisements.

We have over 100,000 in circulation since 1975 and we are still around and That's more than Anyone else can say. We used to sell hundreds of programs individually, the programs in Volume X were sold for several years at over $10,000, in Volume III for over $6,000 but a few years ago we decided to promote software to the mass public and it was an instant success. We are still several years ahead of our time in our marketing concepts as well as our products and we are going to be making another major change in the market. We are going to offer our programs in cassette form. NOT just one of two programs like everybody else. But a book full of programs for just $9.95.

For Homeowners, Businessmen, Engineers, Hobbyists, Doctors, Lawyers, Men and Women

Vol. I $24.95
Business & Personal Bookkeeping Programs

Bond
Building
Compound
Cyclic
Decision 1
Decision 2
Depreciation
Efficient
Flow
Installment
Interest
Investments
Mortgage
Optimize
Order
Pert Tree
Rate
Return 1
Return 2
Schedule 1

Games & Pictures

Animals Four
Astronaut
Bagel
Bio Cycle
Cannons
Checkers
Craps
Dogfight
Golf
Judy
Line Up
Pony
Roulette
Sky Diver
Tank
Teach Me

A. Newman
J.F.K.
Linus
Ms. Santa
Nixon
Noel Noel
Nude
Peace
Policeman
Santa's Sleigh
Snoopy
Virgin

Vol. II $24.95
Binomial
Chi-Sq.
Coeff.
Confidence 1
Confidence 2
Correlations
Curve
Differences
Dual Plot
Exp-Distri
Least Squares
Paired
Plot
Plotpts
Polynomial Fit
Regression
Stat 1
Stat 2
T-Distribution
Unpaired
Variance 1
Variance 2
XY.

APPENDIX A

Beam
Conv.
Filter
Fit
Integration 1
Integration 2
Intensity
Lola
Macro
Max. Min.
Navaid
Optical
Planet
PSD
Rand 1
Rand 2
Solve
Sphere Trian
Stars
Track
Triangle
Variable
Vector

Vol. III $39.95
Billing
Inventory
Payroll
Risk
Schedule 2
Shipping
Stocks
Switch

Vol. VI $49.95
Mini-Ledger
Payroll
A/R
Inventory
Peprec.
Ledger

Vol. VII $39.95
Chess
Medbil
Wdproc
Utility

Vol. IV $9.95
Bingo
Bonds
Bull
Enterprise
Football
Funds 1
Funds 2
Go-Moku
Jack
Life
Loans
Mazes
Poker
Popul
Profits
Qubic
Rates
Retire
Savings
SBA
Tic-Tac-Toe

Vol. V $9.95
Andy Cap
Baseball
Compare
Confid 10
Descrip
Differ
Engine
Fourier
Horse
Integers
Logic
Playboy
Primes
Probal
Quadrac
Red Baron
Regression 2
Road Runner
Roulette
Santa
Stat 10
Stat 11
Steel
Top
Vary
Xmas

APPENDIX B

Vol. VIII $19.95
1040-Tax

Balance

Checkbook

Instol 78

Deprec 2

APPENDIX C

Favorites

Vol. IX $19.95*
Auto
Cypher
Hurrtrac
ID
Lorana
Map
Navigate
Omega
Patterns
Radar
RDF

Vol. X $69.95*
Intro.
A/R
A/P
Mer Inv
Check
Assets
Payroll
Bal Sh
P/L
Year End
Data Base
Tax Up
Basic St.

*10% Discount — Offer good til April 30, 1980

Volume III—Part 1—Billing Inv. Pyrl.
Volume VII—Part 1—Chess
—available on TRS-80 level II compatible cassette $9.95
Add $1.00 per tape Post./Handling

Volume VI — Disk programs are compatible with TRS-80 disk basic
The disk programs in Volumes VI, VII and X are written in (CP/M) M Basic and Disk Extended Microsoft Basic. Other programs written in 8K Basic.

Add $1.50 per volume handling, all domestic shipments sent U.P.S. except APO and P.O. Box which go parcel post. Foreign orders add $6.00/volume for air shipment and make payable in U.S. dollars only.

AVAILABLE AT MOST COMPUTER STORES
Master Charge and Bank Americard accepted.
Our Software is copyrighted and may not be reproduced or sold.

Unlike others we have NOT raised our prices in five years

KEMCO, LTD. ✓26
P.O. Drawer 2208G Petersburg, VA 23803

24 hour Hot Line (800) 528-6050
ext. 1184. In Arizona (800) 352-0458

OVER 116,000 IN USE TODAY

Allan Redstone
24121 Stratford
Oak Park MI 48237

Analog-to-Digital Conversion

For less than $5, the author built an A/D converter for his COSMAC system.

Like all red-blooded American computer hobbyists, I salivate at the sight of ads for peripherals in each new issue of *Microcomputing*. However, as a student on a limited budget, I become disheartened as my eyes wander to the bottom of the page where they invariably find a dollar sign followed by lots of numbers. This situation kept my COSMAC 1802 system small, but one peripheral I felt I had to have was an analog-to-digital converter. Therefore, I set out to make my own, and by substituting software for some expensive hardware, I was able to build one for under $5.

What Is an A/D Converter?

An analog-to-digital converter is a device that changes an incoming voltage into a binary word. In an 8-bit A/D converter, the output varies between 00 and FF, depending on the input voltage.

Therefore, if a 5 volt input voltage produces the maximum FF output, a 2.5 volt input should produce an output of 80. (See Table 1.) Most A/D converters can be calibrated so that the voltage that produces the maximum output can be changed to fit the user's needs.

How It Works

There are many different methods of A/D conversion, but the one I shall discuss is the successive approximation method. In a conventional successive approximation converter, a shift register outputs a bit to a D/A converter, which converts the binary bit into a voltage, which is fed into an op amp along with the input voltage to be converted.

If the D/A voltage exceeds the input voltage, the op amp's output will go high, and a 1 bit will be stored in another shift register. If there is no output from the op amp, a zero is stored. Both registers are then shifted, and the process continues until the least significant bit is output from the first shift register. The second register then contains a binary word that is equivalent to the input voltage.

The circuit for such a converter must contain two shift

There's me with the whole setup. The perfboard just to the left of the enclosed keyboard is the A/D circuitry. On the screen is an example of what can be done with the etch-a-sketch program and a little practice.

Hex	Voltage
00	0.00
10	0.32
20	0.63
30	0.95
40	1.25
50	1.57
60	1.88
70	2.19
80	2.50
90	2.81
A0	3.13
B0	3.44
C0	3.75
D0	4.06
E0	4.38
F0	5.69
FF	5.00

Table 1. An A/D converter produces the hex output when given the corresponding voltage.

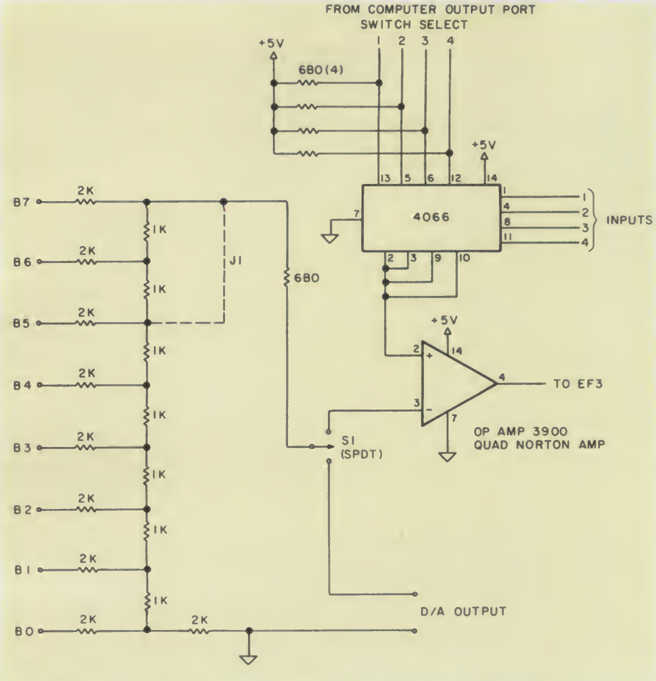


Fig. 1 B0 through B7 are connected to the computer's output port, B0 being the least significant bit. Resistors can be ½ or ¼ Watt types with 10 percent or less tolerance. When using the circuit for etch-a-sketch, short out the two most significant resistors in the resistor ladder by adding jumper J1 (dotted line) and connect switch-select leads 1 and 2 to computer output bits B6 and B7. When using only one input voltage, apply + 5 V to all four switch-select leads.

Close-up view of a picture drawn with the Etch-a-Sketch program.

registers and a clock, all three of which are unnecessary if the converter is to be used with a computer. The computer can output the necessary bits to the D/A converter and decide whether a 1 or a 0 should be stored by sensing the op amp's output. This is precisely what my circuit does.

Hardware

The converter is a combination 8-bit A/D and D/A converter (switch selectable). Using a COSMAC 1802-based system with a 1.7 MHz clock, the A/D circuit is capable of performing over 1000 conversions per second.

I built the circuit in Fig. 1 on perfboard using standard components purchased at a local Radio Shack. It consists of a resistor ladder, an op amp and a quad analog switch. The resistor ladder serves as an 8-bit D/A converter whose output is fed to the op amp along with the input voltage.

By outputting the correct code to the quad switch, the computer can select up to four different inputs. The output of the op amp is connected to the 1802's EF3 line, which can be sensed by the computer.

Software Control

The software for the system is concise, as shown by the flowchart in Fig. 2. The computer first outputs an 80, then tests the EF3 line. If the EF3 line is high, the hex number 80 will be added to a previously cleared register. The 80 is then shifted right and the new number, 40, is added to the register.

The contents of the register (now C0) are output, and EF3 is again tested. If EF3 is low, 40 is subtracted from the register, if it is high, the register is left alone. The 40 is then shifted and added to the register.

This process continues until the carry flag goes high (each shift occurs in the accumulator, the D register in the 1802, and after each shift, the carry flag is tested), at which time all bits have been tested. The register then contains the correct binary word (Program 1).

Applications

Some of the uses for an A/D converter include a digital voltmeter, digital thermometer or even a speech-recognition system. However, I bought my COSMAC system with pixie graphics for games, and cer-

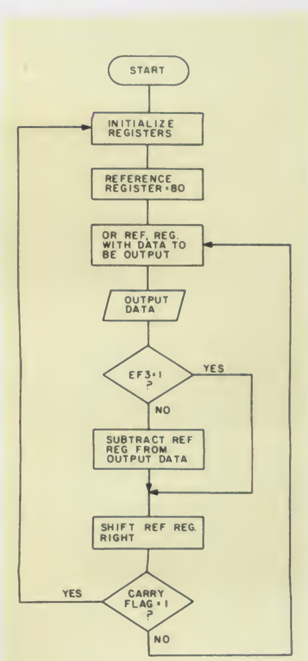
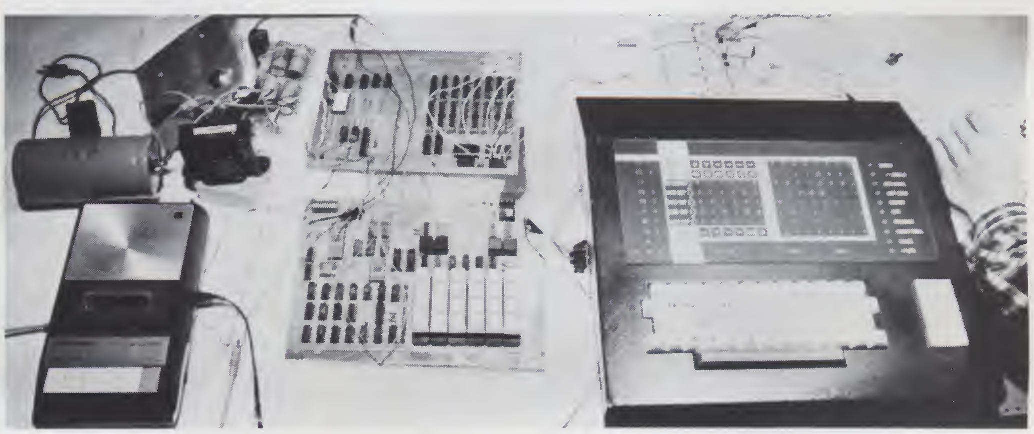


Fig. 2. A/D program flowchart.



My entire system. The two circuit boards in the center are the COSMAC Super Elf computer and 4K memory board. The perfboard just above the enclosed keyboard is the A/D circuit.

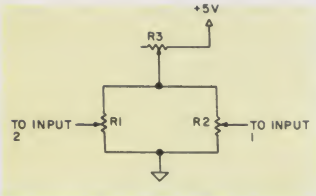


Fig. 3. Etch-a-Sketch potentiometer circuit.

tain games cannot be played without potentiometers for controls. By having a potentiometer vary the voltage to the A/D converter, I can manually "move" objects across my video display.

An example of such a game is the famous Etch-a-Sketch game, which in the electronic version described here consists of two potentiometers: one controls the vertical direction; the other, the horizontal. As you turn the controls, lines are drawn on the TV display. By turning both controls at the same time, you can make diagonal lines (Program 2).

The program takes data from the two potentiometers and converts it into Cartesian coordinates, which are plotted on the TV screen. (COSMAC pixie graphics have a resolution of 32 vertical by 64 horizontal pixels.) The program also clears the screen each time the program is run.

When only one input voltage is being used, the quad switch can be bypassed. However, when two inputs are used, such as in Etch-a-Sketch, the switch must be used, necessitating another output port. Since my system only has one output port, I was confronted with a problem.

After much deliberation, I decided to cut the A/D converter down to six bits and use the last two bits to provide the necessary pulses to the switch. I did this by shorting out the two most significant resistors in the D/A converter, and by outputting a 20 to begin with rather than an 80 (Fig. 1).

The potentiometers used in Etch-a-Sketch are 10k linear taper types (R1 and R2). Unfortunately, it is almost impossible to get potentiometers that are perfectly linear. Therefore, if the vertical control is turned all the way down, it may be necessary to turn the control almost one-quarter before a line forms on the screen. R3 solves this problem by adjusting the voltage into R1 and R2 (see Fig. 3).

In order to calibrate R3, load the program and move the controls all the way down and to the right. Then reset the program to clear the screen, and a dot should appear in the lower right-hand corner. Turn R3 until the dot begins to move, and

```
LOC  INSTRUCTION       COMMENTS

00   90                Locations 00 through 2E contain
01   B1                the video refresh program. Location
02   B2                19 contains the number of the 256
03   B3                byte page to be displayed.
04   B4                The A/D conversion program begins
05   F8  2D            at location 2F
07   A3
08   F8  FF
0A   A2
0B   F8  11
0D   A1
0E   D3
0F   72
10   70
11   22
12   78
13   22
14   52
15   C4  C4  C4
18   F8  02
1A   B0
1B   F8  00
1D   A0
1E   80
1F   E2
20   E2  20  A0
23   E2  20  A0
26   E2  20  A0
29   3C  1E
2B   30  0F
2D   E2
2E   69
2F   30  31            GOTO 31
31   F8  02            R6.0=02  R6.1=02  R4.1=02
33   A6
34   B6
35   B4
36   E6                X=6
37   F8  00            R4.0=00
39   A4
3A   F8  00            D=00
3C   56                Put D into location selected by R6
3D   F8  80            R5.0=80 (Reference data)
3F   A5
40   85                Put contents of R5.0 into D
41   F1                OR D with memory in location selected by X
42   56                Put D into location selected by R5
43   63                Output data in location selected by X
44   26                Decrement R6
45   36  4A            If EF3=1 GOTO 4A
47   85                Put contents of R5.0 into D
48   F5                Subtract D from memory in loc. selected by X
49   56                Put D into location selected by R6
4A   85                Put contents of R5.0 into D
4B   F6                Shift D right
4C   A5                Put D into R5.0
4D   3B  40            If carry flag=0 GOTO 40
4F   06                Put memory in location selected by R6 into D
50   54                put D into location selected by R4
51   30  3A            GOTO 3A
```

Program 1. A/D software. To test the A/D circuit, load the program starting at location 0000. Then apply various voltages to the A/D circuit. The binary equivalent of the voltage should appear on the screen at location 0200.

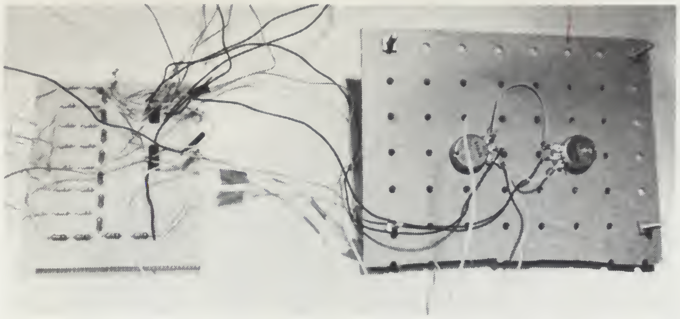
```
LOC  INSTRUCTION       COMMENTS

4F   06                Put data in location selected by R6 into D
50   31  58            If Q=1 GOTO 58
52   54                Put D into location selected by R4
53   8D                Put contents of RD.0 into D
54   56                Put D into location selected by R6
55   7B                Q=1
56   30  3D            GOTO 3D
58   F6                Shift D right
59   5F                Put D into location selected by RF
5A   8E                Put contents of RE.0 into D
5B   56                Put D into location selected by R6
5C   7A                Q=0
5D   30  80            GOTO 80

70   F8  02            RF.1=02
72   BF
73   F8  10            RF.0=10
75   AF
76   F8  80            RD.0=80
78   AD
79   F8  40            RE.0=40
7B   AE
7C   30  D0            GOTO D0

80   F8  03            R7.1=03
82   B7
83   0F                Put data in location selected by RF into D
84   FE  FE  FE        Shift D left three times
87   A7                Put D into R7.0
88   30  C0            GOTO C0
8A   F6  F6  F6        Shift D right three times
8D   54  54            Put D into location selected by R4
8F   54  54
91   E4                X=4
92   87                Put contents of R7.0 into D
93   F4                Add D to memory in location selected by X
94   A7                Put D into R7.0
95   88                Put contents of R8.0 into D
96   FE  FE  FE        Shift D left five times
99   FE  FE
9B   F6  F6  F6        Shift D right five times
9E   F6  F6
A0   A8                Put D into R8.0
A1   F8  80            R9.0=80
A3   A9
A4   88                Put contents of R8.0 into D
A5   32  AC            If D=0 GOTO AC
A7   28                Decrement R8
A8   89                Put contents of R9.0 into D
A9   F6                Shift D right
AA   30  A3            GOTO A3
AC   E7                X=7
AD   89                Put contents of R9.0 into D
AE   F1                OR D with memory in location selected by X
AF   57                Put D into location selected by R7
B0   E6                X=6
B1   30  3D            GOTO 3D
C0   04                Put data in location selected by R4 into D
C1   FE  FE            Shift D left twice
C3   F6  F6            Shift D right twice
C5   A8                Put D into R8.0
C6   30  8A            GOTO 8A
D0   F8  03            RC.1=03
D2   BC
D3   F8  00            RC.0=00
D5   AC
D6   F8  00            D=00
D8   5C                Put D into location selected by RC
D9   1C                Increment RC
DA   8C                Put contents of RC.0 into D
DB   3A  D6            If D does not equal 00 GOTO D6
DD   30  31            GOTO 31
```

Program 2. Etch-a-Sketch program. Load Programs 1 and 2; parts of this program will go over Program 1. In Program 1, change data in location 19 to 03, data in location 3E to 20 and data in location 30 to 70.



Close-up view of the A/D converter and potentiometer assembly. The pots mounted on the pegboard are R1 and R2.

then turn it back slightly. Calibration of the system is complete at this point.

Other Systems

The A/D converter can be used with any computer with a latched output port. If your CPU doesn't have the equivalent of an EF3 line, you can substitute a single bit of an input port. This will slow down conversion time slightly, since additional software must be used to sense the input port.

Wrap-up

The hardware and software described in this article can be used with any COSMAC 1802 system with pixie graphics and parallel I/O, and can be modified for use with virtually any computer. The Etch-a-Sketch program can easily be written for any computer that can plot Cartesian coordinates and a video display. Perhaps most important, the entire A/D circuitry can be built for under $5.■

EXATRON STRINGY FLOPPY Owners Association Newsletter

Secretary, Fred Waters

THE EXATRON STRINGY FLOPPY

For a long time users of microcomputers have had to put up with the delays and reliability problems of cassette tape, or else come up with around $1000 for a disk system, not to mention another $200 for systems software. It has always seemed rather unfair that the microcomputer owner should have to spend more than twice the value of his basic machine on peripherals.

Exatron made a remarkable breakthrough two years ago when it introduced the first Stringy Floppy—the S-100 bus version. During the past year, the popularity of the TRS-80 version has skyrocketed; it offers users a quantum leap in speed and reliability at less than a quarter the cost of an expansion interface and disk.

Now, for the benefit of users of the SS-50 bus with the 6800 microprocessor, Exatron accomplishes another breakthrough. For under $500 the 6800 user may acquire a complete software and hardware package consisting of TWO Stringy Floppy drives, a Controller Board, the new and exciting SIMPLEX-68 Operating System, a BASIC with full data file capability, the Technical Systems Consultants (TSC) Text Editor, and the TSC Assembler. Along with this package comes a box of ten 20-foot wafers, a box of ten 50-foot wafers, and a complete set of documentaiton. At less than half the cost of a dual-drive minifloppy system, this package offers a complete mass storage system capable of handling over 130K bytes on the two drives, a versatile and easy-to-use file manager, utilities, and all of the basic building blocks of a complete software library. Aside from the benefit of a terrific price, the reliability of the Stringy Floppy over cassette techniques is vastly improved, and load times are reduced to seconds rather than minutes.

SIMPLEX-68

The catalyst that has made this total package possible is the SIMPLEX-68 Operating System. SIMPLEX-68 was designed to provide the power, versatility, and flexibility of a disk operating system to the 6800 microcomputer user with a Stringy Floppy. Through the SIMPLEX-68 OS, the user has access to a BASIC with powerful data file capabilities, and an assembler and text editor which no longer require that programs be totally resident in RAM. SIMPLEX-68 consists of three major parts: the operating system, the file management system, and system utilities. The operating system provides the command analysis and operator interface to the terminal. The file management system controls all I/O to the ESF drives and maintains directories for information stored on wafers. The user doesn't have to keep track of where programs reside. Finally, the utilities consist of a set of independent programs which call upon the operating system and file manager in order to perform tasks which manipulate the files and modify and display system parameters. One of the utility functions in NEWTAPE, which will format and verify a new wafer. Another utility program, CAT, provides a full listing of tape directories; LIST displays a text file on the terminal or printer. Other utilities include COPY for copying files from one drive to another, APPEND for joining multiple text or binary files, RENAME, and SAVE.

The complete library of software support has been made possible by making several of the interfaces compatible with the standards used in the TSC FLEX Disk Operating System for the 6800. The OS package includes the TSC Editor and Assembler; TSC Cassette BASIC with ESF LOAD and SAVE commands is available at additional cost. Although TSC BASIC has no data file capability, its unmatched execution speed makes it highly useful for fast action games.

SYSTEM REQUIREMENTS

The SIMPLEX-68 Operating System requires 5K of RAM from $6C00 to $7FFF for system variables and directories. It is recommended that the system be configured with RAM from $0000 to $3FFF in order to run BASIC, the editor, or the assembler. The Operating System itself is provided in EPROM on the ESF Controller Board at $C000. Finally, the SIMPLEX-68 Operating System has I/O interfaces making it easy to use SWTBUG or other monitors with input at $E1AC and output at $E1D1.

A complete software/hardware operating and mass storage system for under $500?!! Through the unique Exatron Stringy Floppy, users can have the speed, reliability and storage capacity of a minifloppy system at a fraction of the cost. Things to come include a version of SIMPLEX for the 6809, and business applications software. Call or write Exatron for further information.

5TH COMPUTER FAIRE

As you are reading this, the Fifth West Coast Computer Faire is about to go on, or has just gone by. We in the SF Bay Area contingent of the Exatron Stringy Floppy Owners Association hope you have or had a chance to come by the Exatron Booth. Five micros of various persuasions, each with a Stringy Floppy doing something wonderful, and enthusiastic members of ESFOA to tell you about them! One of the remarkable things to be seen is the sampling of fine software that ESF owners have developed and have made available. If you don't make the Faire, and want to know more about the Stringy Floppy for either the TRS-80, the S-100 bus, or the SS-50 bus, use the toll-free number below to ask for the very fine information packet.

INFORMATION & ORDERS

The ESF is assembled and tested at the factory, with a 30-day moneyback guarantee and a one-year full warranty. For the fastest delivery, phone in your credit card or COD order using the toll-free line below.

Base price for the TRS-80 ESF, $249.50 (ask about the Starter Kit); for the S-100 ESF, $289.50; and for the SS-50 package above with users manual, $499.00. Users Manuals for the TRS-80 and S-100 versions, and complete information packets for all versions of the ESF are available at no charge. Add $3.00 for handling.

If you have any questions about the product, about Exatron, or ESFOA, please call the Hot Line. Address letters to ESFOA, 3559 Ryder St., Santa Clara, CA 95051.

 ✓7

HOT LINE	800-538-8559
WITHIN CALIFORNIA	408-737-7111

EXATRON STRINGY FLOPPY
WORKSHOP CHAIRMEN

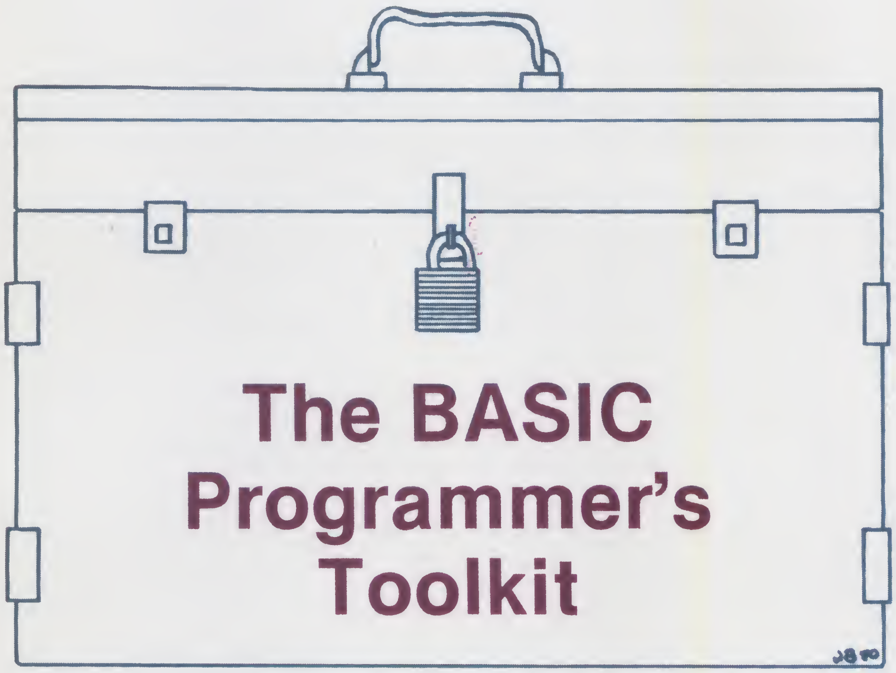
WORKSHOP CHAIRMEN (BY ZIP CODE)

Jack O'Connell, 7 Royal Crest Drive, Marlboro, MA 01752 (617) 481-2417
Austin McCollough, 24 Donna Rd., Chelmsford, MA 01824 (617) 256-0473
R.M. Knowles, M.D, 88 Foreside Rd., Cumberland Foreside, ME 04110 (207) 773-7261
Wesley R. Day, 86 Merryfield Ave., Waterville, ME 04901 (207) 873-3955
Rev. Richard R. Noyes, 118 Main Street, Madison, ME 04950 (207) 696-5929
Michael Keller, RFD 1, Box 507, Solon, ME 04979 (207) 643-2437
Thomas Penisch, 310 Palisade Ave., Bogota, NJ 07603 (201) 343-3828
Ted Kingston, Willard Rd., Strathmere, NJ 08248 (609) 263-2440
Milton F. Thrasher, 8C Fairway Ave., Rye, NY 10580 (914) 967-3322
John E. Sheil, 196 Blacksmith Rd., Levittown, NY 11756 (516) 731-6467
HYNX Company, P.O. Box 577, Newtonville, NY 12128 (518) 783-9495
Bill Firm, 244 Albany Ave., Kingston, NY 12401 (914) 383-1824
George C. Hadley, Box 10, Paul Smiths, NY 12970 (518) 327-6413
Richard C. McGarvey, 221 Hirschfield Dr., Williamsville, NY 14221 (716) 634-3026
Robert C. Wynne, P.O. Box 704 F.S., Niagra Falls, NY 14303 (716) 285-9391
Gary E. Snyder, 37 Sumner Park, Rochester, NY 14607 (716) 461-1364
David L. MacAdam, 68 Hammond St., Rochester, NY 14615 (716) 865-8917
Paul Sturpe, 220 Wessex Hills Dr., Coraopolis, PA 15108 (412) 264-1492
Bill Dalesandry, 355 Rockfield Rd., Pittsburgh, PA 15243 (412) 221-1968
Carl A. Kollar, 1202 Gemini St., Nonticoke, PA 18634 (717) 675-5377
Bruce A. McIntyre, Rd 1, Bridgetown Pike, Langhorne, PA 19047 (215) 322-1895
Richard S. Landy, 1008 Evergreen Rd., Morrisville, PA 19067 (215) 736-0564
Robert G. Kelch, 23 Mapledale Ave., Glen Burnie, MD 21061 (301) 760-4793
Charles A. Sutton, 213 Marshall St., Salisbury, MD 21801 (301) 742-9318
Richard Harrison, Rt. 2 Drysdale, Warrenton, VA 22186 (703) 439-8109
William P. Schultze, 931 Branchwater St., Fredericksburg, VA 22401 (703) 786-8878
William C. Geer, Rte 5, Box 502, Falmouth, VA 22401 (703) 371-2019
H. Ross Wiant, 240 Brook La., Grafton, VA 23692 (804) 898-8120
Archie H. Hubbard III, 520 Lawrence Ave., Bristol, VA 24201 (703) 669-3191
Lawrence P. Doyle, 717 Patrick St., Eden, NC 27288 (919) 627-0359
Micalel W. Talbot, 3601 Carriage Dr., Raleigh, NC 27612 (919) 787-8082
Herbert Alperin, 5675 Peachtree Dunwoody Rd NE, Atlanta, GA 30342 (404) 256-3414
Mark C. Few, 2827 Parkland Cir., Orlando, FL 32803 (305) 644-7464
John T. Charlton, 688 Caribbean Rd., Sat Beach, FL 32937 (305) 773-1590
Richard B. Nelson, 763 W. 63rd St. Ocean, Marathon, FL 33050 (305) 743-9195
Frank Lungarella, 648 DeLeon Dr., Miami Springs, FL 33166 (305) 887-4821
A.J. Catano, 11810 NW 29th Pl., Sunrise, FL 33323 (305) 741-4325
Robert W. Hopkins, 212 31st St., West, Bradenton, FL 33505 (813) 748-7074
William C. Howes, 5597 Trellis Ln., Fort Myers, FL 33907 (813) 482-0347
Bob Jones, 4084 Berrybrook Cove, Memphis, TN 38138 (901) 363-5469
Sumner Loomis, Rt. 1, Box 131, Prairie Point, MS 39353 (601) 726-5524
Lawrence W. Stein, 867 Independence Station Rd., Independence, KY 41051 (606) 371-3242
W.H. Jenkins, 33 W. Columbus St., Canal Winchester, OH 43110 (614) 837-2325
Lester R. Snowden, 1821 Pine St., Rafb, OH 43217 (614) 497-0838
Paul Schmid, 201 Mohican Ave., Orrville, OH 44667 (216) 682-3596
Robert E. Morgan, 1127 Princeton, Celina, OH 45822 (419) 586-7416
Don Heringhaus, 1055 Woodside Dr., Ottawa, OH 45875 (419) 523-5613
William R. Gouge, 6428 Topeka Ln, Indianapolis, IN 46241 (317) 856-6339
Jack J. Fleming, 322 Jeffrey, Royal Oak, MI 48073 (313) 435-4487
Robert M. Clark, 20515 Gaylord, Redford, MI 48240 (313) 535-5722
Dennis Chaney, 1239 S. Ruess, Owosso, MI 48867 (517) 723-1011
Alan C. Lanphear, 727 Ray St., Kalamazoo, MI 49005 (313) 344-8277
Ranjit Diol, RR6, Kilburn Ave., Dowagiac, MI 49047 (616) 424-5440
Donald Meinke, 2507 Blaine SE, Grand Rapids, MI 49507 (616) 245-9420
R.L. Knapp, Jr., 414 2nd St., Perry, IA 50220 (515) 465-5873
Robert F. Gerald, 1377 Arden View Dr., Arden Hills, MN 55112 (612) 636-0549
Robert W. Foster, 791 White Birch Ln, Lake Zurich, IL 60047 (312) 438-5223
Martin Berger, 4634 Oakton St., Skokie, IL 60076 (312) 674-3038
Joel Natkin, 1626 Keeney, Evanston, IL 60202 (312) 869-1660
Gary P. Cocanour, 333 Central Ave., Hinsdale, IL 60521 (312) 986-1059
James E. Spath, 3610 N. Pinegrove Apt. 301, Chicago, IL 60613 (312) 353-2544
Richard Kramer, 401 N. Cherry, Abingdon, IL 61410 (309) 462-2859
John M. Delaney Jr., 860 Penning Ave., Wood River, IL 62095 (618) 254-4571
Donald L. Hiles, 802 Grand Ave., Beardstown, IL 62618 (217) 323-3963
Lynn A. Nelson, 409 San Jose, O'fallon, MO 63366 (314) 272-1153
James G. Runyan, 1400 N. Pine Apt. 6, Rolla, MO 65401 (314) 364-7306
Ed Forest, 8224 Tomahawk Rd., Prairie Village, KS 66208 (913) 648-1513
Charles L. Perrin, 5001 E. Pawnee 113B, Wichita, KS 67218 (316) 685-0528
Val Lindholm, Box 1105, 5512 Comanche, Great Bend, KS 67530 (316) 793-9319
Robert K. Lahm, 224 4th Ave., Holdrege, NE 68949 (308) 995-4071
R.M. Wright, 3416 Greenbriar Ln, Plano, TX 75074 (214) 596-7000
Dr. Murlon H. Dye, 1412 Locust St., Commerce, TX 75428 (214) 886-6434
Donald Michel, 2601 Rushvalley Ct., Arlington, TX 76016 (817) 469-1521
W.L. Lonnon, Rt 3, Box 49-1, Decatur, TX 76234 (817) 627-3838
Kent Zakrzewski, 619 Cortlandt, Houston, TX 77007 (713) 448-6374
George Lyda, Box 127, New Baden, TX 77870 (713) 828-4523
Stephen E. Fisch, 5225 Moultrie, Corpus Christi, TX 78413 (512) 991-4584
R. B. Sexton, 2200 W. Anderson Ln., Austin, TX 78757 (512) 452-6131
John Dierdorf, 8109 Greenslope Dr., Austin, TX 78759 (512) 345-3912
Richard Laubhan, 111 N. Claremont, Colorado Springs, CO 80909 (303) 632-7580
Dennis E. Murray, 989 E. 1050N. Ogden, UT 84404 (801) 782-8042
John Crutcher, 6015 E. Pershing Ave., Scottsdale, AZ 85254 (602) 948-2911
Bob Rankin, 1018 W. Hermoja Dr., Tempe, AZ 85282 (602) 967-7677
Robert D. Sullivan, 1537 W. 7th St. No. 120, Upland, CA 91786 (714) 985-5273
Frank M. Harris, 4570 Ohio St., San Diego, CA 92116 (714) 281-9286
Donald Haskell, 4404 W. Hurley Ave., Visalia, CA 93277 (209) 732-1553
Hubert C. Wood, 201 McCord, Sp. 5, Bakersfield, CA 93308 (805) 399-8252
Bill Burnham, 3659 Glenwood Ave., Redwood City, CA 94062 (415) 365-0760
G. K. Johnson, 1009 Aspen Way, Petaluma, CA 94952 (707) 762-2168
Dustin F. Leer, Box 1664, Sausalito, CA 94965 (415) 332-2557
James Ferguson, 2485 Autumnvale Ave., San Jose, CA 95132 (408) 946-1265
Lars A. Benson, 436 Klute St., No. 2, Santa Rosa, CA 95401 (707) 523-1782
Richard D. Mack, 1748 Fenwick Dr., Santa Rosa, CA 95401 (707) 528-0309
J. J. Waters, 333 LaCrosse Ave., Santa Rosa, CA 95405 (707) 539-2595
Vaughn A. Jupe, RFD Box 60, Carlotta, CA 95528 (707) 768-3833
E.R. Hubbard, 2558 Deertrail Ln., Shingle Springs, CA 95682 (916) 677-5440
Paul F. LeFevre, 3529 Cody Wy 109, Sacramento, CA 95825 (916) 483-4760
Alfred J. Kiep, Jr., 605 Wilcox, Ave., Etna, CA 96027 (916) 467-3255
Gilbert Olayan, 961 Apokula St., Kailua, HI 96734 (808) 261-2228
L.P. Folz, 419 Madrona SE, Salem, OR 97302
Ken Ernst, 3921 Ibex NE, Salem, OR 97303 (503) 393-1173
J.R. Shunn, 2625 Altamont No. 10, Klamath Falls, OR 97601 (503) 882-2023
Rick Winter, P.O. Box 933, Pendleton, OR 97801 (503) 276-3611
J.A. Records, 1106 Cedar St., LaGrande, OR 97850 (503) 963-0437
Ralph Baker, P.O. Box 250, Long Beach, WA 98631 (503) 642-3126
Allan Stubbs, 1598 Cedar St., Prince George, B.C. Canada, V2L-1B7 563-6894

6800 SS-50 ESF OWNERS

John Drum, 44 County Village Ln., Sudbury, ME 01766
John Pyra, 44 Flagg Rd., Westford, ME 01866
James Robertson, 21 Grafton Dr., Morris Plains, NJ 07950
G.S. Lyons, 217 Homestead Ave., Huddonfield, NJ 08033
Bits Computer Systems, 337 Geln Rd., Rome, NY 13440
Richard Kerns, 616 Beatty Rd., Monroe, PA 15146
Mike Mellon, 211 San Paulo Cir, W. Melbourne, FL 32901
Norman Weiner, 1700 Convention Center, Miami Beach, FL 33139
Gordon Doughman, 4250 Moselle Dr., Hamilton, OH 45011
Paul Manter, 10665 Stargate Ln., Cincinnati, OH 45240
Garrett Smith, 2001 NW Hickory Ln, No. 9, Ankery, IA 50021
Mark Begemann, 1985 24th St., Mario, IA 52302
Ray Westberg, 5702 34th Ave., N. Minneapolis, MN 52422
Rodney Schnueland, 9027 Pierce St. NE, Blaine, MN 55434
Laurence Drymalski, 1232 Countryside Dr., Elgin, IL 60120
Donald Miser, 1015W Wood, Bloomington, IL 61701
John Ramsey, 8307 E. 166th St., Belton, MO 64012
Al Scherer, M.D., 610 E. 1st St., St. John, KS 67576
Wade Maxfield, 2525 E. Union Bower, Evering, TX 75061
Richard Poindexter, 4347 Phelan St., Beaumont, TX 77707
John Stimm, 6220 Calebra Rd, San Antonio, TX 78284
European Auto Parts, 4322 B.N. Lamars, Austin, TX 78756
John Lutenberg, 923 Main St., Canyon City, CO 81212
UC Los Alamos Scientific Lab, Los Alamos, NM 87545
Thomas Mattingly, 1005 Essex Dr. W. , Las Vegas, NV 89107
Robin Ellwood, 18506 Dylan St., Northridge, CA 91326
Don McPhee, 1550 4th Ave., S., Seattle, WA 98134
P. Peterson, 9117 NE 96th St., Vancover, WA 98662
Jerry Starzinski, 58th Ave., Yakima, WA 98903
Motorola, 2400 S. Rossevelt, Tempe, AR
J.L. Bauliane, 956 Lillian St., Coquittam, British Columbia, CAN V3J 5C4

Tom Hayek
723 Hialeah Dr.
Racine WI 53402



The BASIC Programmer's Toolkit

Lift the lid on Nestar's toolkit and have a look inside.

In August 1979 I first noticed advertising descriptions of the BASIC Programmer's Toolkit, which claimed to add ten more commands to the Commodore CBM BASIC vocabulary. The commands are: AUTO, DELETE, RENUMBER, APPEND, DUMP, TRACE, STEP, OFF, FIND and HELP. Since the commands are in firmware (ROM), they will not take up any of my 32K of RAM; they will be there whenever I turn on the computer.

I was eager to have this additional command capability and placed my $49.95 order for the ROM, which is a product of Palo Alto ICs, a division of Nestar Systems, Inc.

After a short delay, my Toolkit arrived by first-class mail. I eagerly opened the padded mailer to find the ROM placed in a conductive plastic pin carrier, protecting it from the possibility of bent pins and electrostatic damage. Also enclosed was a 34-page book of documentation, professionally printed with a firm, slick cover.

I immediately got the impression that this company cared about what their customers thought about them. At this point I was favorably impressed and developed some confidence in the product and the company.

Installation

The documentation begins with a brief description of the commands and an explanation of the Toolkit installation, followed by a clear and complete description of each command, including examples. Installation in my 32K CBM with a full-size keyboard is a simple matter of opening the case and installing the 2K ROM in one of the three existing empty sockets. Installation on an 8K PET involves plugging a small board containing the ROM and some address decoding into the memory expansion port on the side of the PET. Another small connector with a single wire plugs into the second cassette port to supply 5 V dc to the board.

As the Toolkit manual warns, turn off the power and disconnect the computer from the ac line. Ground yourself by touching the metal case of the computer to dissipate any static charge just prior to handling the Toolkit for installation.

The ROM occupies memory positions B000–B7FF (hex), so if you have other memory expansion systems, such as Skyles or ExpandaPet, be sure to find out—from their respective documentations—the correct socket to plug into. Also, be sure to have the correct orientation of the ROM with respect to pin one.

I installed the ROM and checked for any bent pins. I was now ready to check it out. After powering up my CBM, I entered a BASIC command SYS 45056. This initializes the Toolkit ROM, and the CRT should read: (C) 1979 PAICS. Everything went smoothly, and I was ready to explore all the BASIC commands now available from the Toolkit in the direct mode.

The Commands

AUTO—provides automatic line numbering as you type in your BASIC program. The general syntax is:
AUTO beginning line number, interval.

If you type in AUTO without specifying any parameters, line numbering will start with 100 and the interval will be 10. To get out of the AUTO mode, just hit the return key without entering anything after the line number.

AUTO also remembers where it left off. If you exit the AUTO mode to do some editing and then type AUTO, numbering will start at the next sequential line in your program. The previously set interval will be maintained. If you type AUTO 200, line numbers will start with 200 and be incremented by the last interval given in the AUTO command.

AUTO helps to ease some of the typing drudgery in entering a BASIC program.

RENUMBER—renumbers the entire BASIC program presently in memory. All GOTO, ON . . . GOTO, GOSUB, ON . . . GOSUB, IF-THEN, RUN and LIST commands are also changed to the new respective reference line. All references to nonexistent line numbers are changed to 63999. This is especially useful when used with the FIND command. The general syntax is:
RENUMBER beginning line number, interval.

If you type RENUMBER without specify-

ing any parameters, renumbering will start with line 100 and the interval will be 10. It took about 30 seconds to renumber a 10K program.

DELETE—removes BASIC lines by specifying the line number or range of line numbers in the same way that the PET/CBM LIST command lists lines. For example, DELETE 50 deletes line 50; DELETE 50-100 deletes lines 50 through 100; DELETE -100 deletes all lines from lowest through 100; and DELETE 100- deletes all lines from 100 through highest.

The Toolkit is designed so that if you type DELETE without giving a range or specific line number you will get SYNTAX ERROR?. This prevents the loss of the entire program by mistake.

APPEND—will load a program from a cassette and add it to the end of a program already in RAM. It works in the same way as the PET/CBM BASIC command LOAD. The general syntax is:
APPEND "program name," cassette drive (1 or 2).

As with the PET/CBM LOAD command, no specification of the cassette drive defaults to cassette drive #1.

APPEND is convenient for adding previously written subroutines to a program in RAM. You could have several often-used subroutines stored on tape and APPEND them to an existing program under development in RAM.

Caution: You must keep the line numbers in order. APPEND will add anything on the tape to the end of the program in the computer. It is a good idea to number all of your subroutines in the 60000-63000 range and not use this range for your BASIC main body programs. This will help to avoid conflicts in duplicate line numbers when appending.

FIND—locates and displays all lines that contain a specified BASIC keyword, section of a BASIC statement or a quoted string constant. The general syntax is:
FIND BASIC code, line number-line number
FIND "string", line number-line number.

The line-number-parameter-search range performs the same as the PET/CBM LIST command range and the Toolkit DELETE command range. If you omit the line number parameters, the whole program will be searched.

FIND allows you to be as specific as necessary when detailing the BASIC statement or string parameters. For example, FIND FOR I will locate and list every line containing FOR I; FIND A will locate and list every line *containing* the variable A; FIND "THIS" will locate and list every line containing the word THIS; FIND GOTO 100, 10-20 will search lines 10 through 20 and list all lines containing GOTO 100.

As you can see, this proves to be a valuable time-saver. Recall in the description of RENUMBER that any references to nonexistent line numbers are assigned a value of 63999. Now we can use the statement FIND 63999 to list any bad references in the program.

When you use FIND, the number of lines listed on the CRT may be sufficient to cause scrolling. You may slow down the scrolling by holding down the RVS key or stop it anywhere with the STOP key.

DUMP—displays all the non-array variables in memory. They are displayed in the form: variable name = present value (i.e., A = 2). This is a great help in debugging programs. Putting STOP statements in the program and then checking the variables at that point is one way to find out where the program is amiss.

DUMP may fill the CRT and cause scrolling. This can be stopped by holding down the SHIFT key. Releasing the SHIFT will allow the scrolling to continue. The STOP will cause the scrolling to stop and abort, as in FIND.

HELP—When you encounter an error while running a program, the PET/CBM will stop the program and print an error message and line number. The HELP command will list the line and indicate the error within the line with a reverse field cursor. The syntax is: HELP.

HELP must be executed before anything else after an error message, otherwise the source of the error will be lost. In that case, executing HELP will do nothing and the computer will come back with READY.

The cursor is usually placed on the character just before the error, but in some cases will be on the error. In the case of 10 B = A / 0, the cursor would be on the 0 (division by 0 is an error).

TRACE—turns on a tracer mode, which will display the currently executed line number when the program is running. The last six line numbers are visible in a reverse field window printed in the upper right-hand corner of the CRT. These six lines scroll from bottom to top within the window, with the most recent line number at the bottom.

Pressing SHIFT will slow the program and scrolling down to about two lines per second.

STEP—also activates the tracer mode, executing one line of BASIC at a time. The line numbers and reverse field window appear just as in TRACE. To execute the next line, momentarily press SHIFT. If you hold the SHIFT key down, the program will continue to run until SHIFT is released. To stop, simply press STOP.

STEP can be conveniently used in debugging also. Being able to single step through a suspected problem area aids in locating the possible faulty coding.

OFF—turns off either the TRACE or STEP commands.

Types of Toolkits

There are basically two types of Toolkits: a 2K ROM that plugs into an empty socket in the new PET/CBM (16K/32K) or an expansion board such as BETSI, and the ROM and an interface IC mounted on a small PC board that plugs into the memory expansion port on the 8K PET. This board has a single wire with a small connector that plugs into the second cassette port to supply 5 V dc for the board.

The costs of the two types are $50 and $80, respectively. The Toolkit comes with a money-back guarantee if you are not completely satisfied; there is also an exchange policy. If you purchase a Toolkit for a PET with the old ROM set and then decide to update to the new ROM set, you can exchange your Toolkit for one that will work with the new ROM set for $15.

Conclusion

Palo Alto ICs and Nestar Systems are not mail-order houses. Do not try to order from them, as you will only delay in getting your Toolkit. You should order from your local computer store. The only mail-order firm that I have seen advertising the Toolkit is Skyles Electric Works, 10301 Stonydale Dr., Cupertino CA 95014. ■

TBS-80
GENERAL ACCOUNTING SYSTEMS. ONE STEP BEYOND.

Your TRS-80™ microcomputer is not a toy. These **TBS-80 general accounting systems** aptly demonstrate the power of your computer.

ANALYSIS PAD by Del Jones is the epitome of first-class programming in business applications. Requiring 48K, and one disk with a printer recommended, this columnar calculator gives the user tremendous flexibility in data entry, enabling the user to create 30 or more columns and rows. Enter your own column and row labels. Enter your data by row or column or directly onto screen display via edit mode. Move, swap, delete, and add rows or columns. Create new pads by stripping relevant data from old files. You never have to key in data twice. But, more important than the powerful data manipulation provided, add, subtract, multiply and divide one column by another and put results in another column. Perform up to six calculations on one column and even define one column to be a constant. The calculation routine you create can be saved and reused. Print out the entire pad in four column segments to line or serial printer. **ANALYSIS PAD** was originally advertised for 32K tape at $32.50. Since then, it has been totally rewritten and expanded to its present 48K disk only form and sells for $49.50. It is easily worth twice as much. You have to see it to believe it.

CHECK REGISTER ACCOUNTING SYSTEM, adapted for the TRS-80 by Dale Kubler and originally written by O.E. Dial, is the most comprehensive check-balancing program written. Requiring 32K, two disks and printer, this program does much more than just balance and reconcile your checkbook. It enables you to define up to 60 account names and will generate monthly summaries of all accounts with monthly and year-to-date totals. Single-entry input allows the user to disperse one transaction over several accounts and to make a 64-character note on each transaction. Checks can be printed out after data has been entered. Aside from the Statement of Accounts, **CRAS** also generates the following reports: Check Register for any Month, Notes to Check Register, Income/Expense Distribution, Statement of Selected Accounts, Bank Reconcile Statement and Suspense File. The Suspense file is an extra feature where you can make notes to yourself for any month in the year. **CRAS** will make both you and your account happy and it sells for $49.50.

CHECKBOOK II by Alan Meyers is the finest program of its kind yet published. With superb graphic screen displays, it does everything necessary to keep your checkbook balanced. Data is input directly into a five-column screen display with a field for alpha or numeric codes. Editing is done easily for changes in any or all columns. **CHECKBOOK II** will accurately balance and reconcile your checkbook, handling balances up to $1,000,000. Your balance brought forward is always in memory. Outstanding checks are listed and easily saved. You can also search for an entry by any field except amount, and all checks with matching entries will be displayed and totaled. A numeric sort routine is included. Screen prints can be made to a line printer from almost any point in the program. In addition, the 32-48K version can write files to disk. This, and the 16K version, are included on the same tape. For $18.50, **CHECKBOOK II** is the bottom line in personal checkbook programs. A disk version of this program is available for $28.50.

BUDGET II (not yet released) by Alan Meyers, takes off where CHECKBOOK II ends. Written exclusively for either disk or tape based computers, this program enables the user to set up 20 account names with four character codes for each, that correspond to the codes used in Checkbook II. Each account can be tagged income or expense and whether it is fixed or not. Set your monthly budget and balance it. Disperse your cash account over the other accounts. Checkbook II data is brought in and summarized by account and compared to amount budgeted. Year-to-date totals are included in monthly summary. Year Summary gives monthly and year totals for each account at a glance. Forecast feature enables user to enter rate of inflation and income increase to see financial standing after 12 months. Review enables user to go back and look at months previously summarized. Flashy graphics and much more. For 16K and 32K tape, **BUDGET II** sells for $24.50. For 32K up disk, $34.50. If you have CHECKBOOK II, you will want this program.

TBS has other incredible software for Tandy's microcomputer. Intent on making it a powerful tool, we have **large scale business accounting systems, data processing systems, system utilities,** and the **Library 100.** We have the only **DISK HEAD CLEANER** (for APPLE too!) and **GRAN MASTER DISKETTES,** the best on the market.

TBS is **YOUR COMPANY,** and we build systems, not just software. The above products are available now, nationwide. Visit your local Computer Dealer or Associate Radio Shack Store and demand the best, demand TBS. For more information, contact us through the numbers below.

™TRS-80 is a trademark of the Tandy Corporation.

TBS™
THE BOTTOM SHELF, INC.
(404)939-6031 • P.O. Box 49014 • Atlanta, GA 30359

✓8

✓Reader Service—see page 257

Programming the Z-80

Using part 1 of this article (a five-point guide to Z-80 programming), which appeared in last month's issue, you're now ready to classify the Z-80's instructions and write a program.

Pat Macaluso
9 Church Court
White Plains NY 10603

In part 1, I introduced you to guidelines and aids helpful in making use of machine language for your application without investing a lot of effort learning machine language or studying and memorizing the instruction set of the Z-80. Now let's look at a method of classifying the instructions to suit our particular application and try our hand at writing a program using the techniques we've learned.

The Z-80 Jungle

A major stumbling block in Z-80 programming for the occasional programmer is the profusion of instruction forms and codes. The Z-80 takes the unused codes from the 8080 processor's one-byte set of possible operation codes and uses them to point to additional codes in the next byte. For some instructions, this scheme is extended to include a third byte. Thus, with additional bytes for an address, for data or for a displacement, a Z-80 instruction may have as many as four bytes.

One way to manage this is to classify the instructions according to a variety of schemes. These include classification by the mode of addressing, of which the Z-80 boasts ten, or by type of processing, such as arithmetic, logical, interrupts, control transfer, bit manipulation. Some of these are further classified as 4-, 8- and 16-bit operations.

On the surface, this appears to make sense, since the user eventually has to deal with such details, or so it seems. In practice, such classifications offered by vendors and textbooks are not too helpful. In fact, they sometimes make the task of using the instructions even harder by forcing the user to deal with machine considerations not necessarily or immediately relevant to the application. The user wants to identify an instruction pertinent to his application, *before* he has to deal with features peculiar to the machine or language.

This is the key to the problem and points to its solution. The key is to classify the instructions by those features that are most immediately useful to the solution of an application problem.

Before we get into this, it should be explained that the disappointing classifications offered the applications programmer by all principal sources of information are not due to any lack of skill or understanding. These sources are, in fact, quite clear and thorough, and their use is recommended.

The difficulty is rooted in a problem that pervades the entire computer field and most of its texts and manuals: most of the literature is written by machine designers and other hardware-oriented types or by systems programmers who concern themselves largely with the relation of hardware to operating systems. Most language designers and implementers have had similar orientations. With rare exceptions, their views of computers and of their use is not that of the application programmer. We users can only take what they have provided—gratefully or otherwise—and must then take care of ourselves as best we can.

A Clearing in the Forest

The particular classification adopted here is shown in Tables 1, 2 and 3. This gets us down to 71 types of instructions. By breaking them up into three categories—register instructions (Table 1), memory instructions (Table 2) and execution control instructions (Table 3), with four subclasses each—they are reasonably manageable.

The real key to this classification is the short, functional description of each instruction. The tables are entered through these descriptions, which match as closely as possible the procedural view of the application programs. Table 4 explains the symbols used in the preceding three tables, along with exceptions.

It is characteristic of machine languages that they make heavy reference to machine features. While we want to play this down, it is unavoidable. The best way out is to face it squarely and accept that the main orientation of the Z-80 instruction set is toward register, memory or execution control operations.

Registers play a central role in microprocessor operations. They can be viewed as very fast, short memories that are accessed by name. They are also involved in all memory operations. Therefore, when a particular function is required, we should first check the possibility of using registers only rather than main memory.

The functional descriptions are in two parts: a short, English descriptive phrase and an algebra-like expression. This may seem like a strange way to achieve clarity, but a little reflection will show that this use of two extremes is the best method. English is easily the language of choice for fast comprehension. But English, like any of the "natural" languages, abounds with ambiguity. Any attempt to be precise would lead

to long, cumbersome statements, thereby losing the advantage of quick comprehension.

The solution to this general problem was arrived at centuries ago by the invention of an alternate to natural language, namely, algebra. While an algebraic expression is not comprehended as rapidly as an English phrase, it is vastly more precise and compact. It thus gets you quickly to the point of explicit working comprehension.

To illustrate the advantage of this combined approach, we will look at a simple example. Suppose we wish to perform an addition using some value in memory. We look in the memory instruction table under arithmetic operations and find the entry "Add memory." That sounds like what we're looking for, but what does it mean? The expression $A \leftarrow A + M$ supplies a concise answer.

The contents of the memory location, for example, the location pointed to by the contents of the HL register pair, are added to the contents of the accumulator, or A-register, and their sum is stored in the accumulator. If these expressions are not clear when first encountered, ignore them for the time being.

Having found the instruction type that we need, we can now zero in on a specific instruction. Depending on your overall objectives, you can make a tentative selection of one of the choices offered and read the full description of the instruction given in Z-80 manuals. Again, I strongly recommend parallel reading.

I generally look first in the assembler manual. If the description and examples are not crystal clear, I look in both the Barden (*The Z-80 Microcomputer Handbook*) and Osborne (*Z-80 Programming for Logic Design*) texts. If the addressing mode, use of registers, timing, etc., meet my program design requirements, I write down the instruction with side notes on registers, flags, etc., as needed. If not, I look at the next most likely instruction in that set.

Soon you will have the basis for a reasonable choice of register allocations, addressing modes, etc. If a change in choices is indicated, then a quick scan of your notes will be helpful. So will a simple block layout of the registers on quadrille paper, on which you can indicate your allocation of data variables.

A particularly useful tool for this kind of quick learning/immediate application is the examination of short Z-80 routines that are fully described. Some excellent ones can be found in Barden's *Handbook*. Once you have written your own routines, document them carefully with a statement of function performed, the meaning of variables and comments on each program line. Save them in an easily retrievable form; they are real programming gold. One good working example of the DJNZ instruction tells the story

Function	Definition	Mnemonic	Operands (opd)	Clock Cycles
Data Transfer				
Copy	$R_L \leftarrow R_R$	LD	R,R	4
		LD	SP,HL	6
		LD	SP,IR	10
		LD	A,R (6)	9
		LD	R,A (6)	9
Swap	$R_L \rightleftarrows R_R$	EX	DE,HL	4
		EX	AF,AF' (8)	4
		EXX	RP,RP' (1) (8)	4
Store Constant	R←N	LD	R,N	7
		LD	IR,NN	14
		LD	RP,NN (5)	
Arithmetic Operations				
Add	$R_L \leftarrow R_L + R_R$	ADD	A,R	4
		ADD	HL,RP (5)	11
		ADD	IR,RP (7)	15
Add Constant	A←A + N	ADD	A,N	7
Add Constant with Carry	A←A + N + F[C]	ADC	A,N	7
Add with Carry	$R_L \leftarrow R_L + R_R$ + F[C]	ADC	A,R	4
		ADC	HL,RP (5)	15
Subtract	A←A – R	SUB	R	4
Subtract with Carry	A←A – (R + F[C])	SBC	R	4
Subtract Constant	A←A – N	SUB	N	7
Subtract Constant with Carry	A←A – (N + F[C])	SBC	N	7
Subtract from Zero	A←1 + ∼A	NEG		8
Increment	R←R + 1	INC	R	4
		INC	RP (5)	6
		INC	IR	10
Decrement	R←R – 1	DEC	R	4
		DEC	RP (5)	6
		DEC	IR	10
Do accumulator arithmetic as BCD		DAA		
Logical Operations				
AND	A←A↑opd	AND	R	4
		AND	N	7
OR	A←A↓opd	OR	R	4
		OR	N	7
Exclusive OR	A←A≠ opd	XOR	R	4
		XOR	N	7
Test bit X	F[Z]←∼R[X]	BIT	X,R	8
Set bit X to 1	R[X]←1	SET	X,R	8
Reset bit X to 0	R[X]←0	RES	X,R	8
Compare	Set Flags	CP	R	4
		CP	N	7
Negate	A←∼A	CPL		4
Negate Carry Flag	F[C]←∼F[C]	CCF		4
Set Carry Flag to 1	F[C]←1	SCF		4
Manipulative Operations				
Shift left, arithmetic	F[C]←R←0	SLA	R	8
Shift right, arithmetic	R[7]→R→F[C]	SRA	R	8
Shift right, logical	0→R→F[C]	SRL	R	8
Rotate left thru Carry	F[C]←R←F[C]	RL	R	8
		RLA	(R = A)	4
Rotate left circular	F[C]←R←R[7]	RLC	R	8
		RLCA	(R = A)	4
Rotate right thru Carry	F[C]→R→R[C]	RR	R	8
		RRA	(R = A)	4
Rotate right circular	R[0]→R→F[C]	RRC	R	8
		RRCA	(R = A)	4

Table 1. Z-80 instructions involving registers only.

Function	Definition	Mnemonic	Operands (opd)	Clock Cycles
Data Transfer				
Read Memory, One Byte	R←M	LD	R,(HL)	7
		LD	A,(RP) (1)	7
		LD	A,(AD)	13
		LD	R,(IR + D)	19
Read Memory, Two Bytes	R←M[AD + 1],M	LD	HL(AD)	16
		LD	RP,(AD)	20
Write Memory, One Byte	M←R	LD	(HL),R	7
		LD	(RP),A (1)	7
		LD	(AD),A	13
		LD	(IR + D),R	19
Write Memory, Two Bytes	(M[AD + 1],M)←R	LD	(AD),HL	16
		LD	(AD),RP	20
Write Constant into Memory	M←N	LD	(HL),N	10
		LD	(IR + D),N	19
Copy Memory, Step-Wise	M_1←M_2	LDD		16
		LDI		16
Copy Memory, Block-Wise	M_A←M_B	LDDR		21/16
		LDIR		21/16
Read Top of Stack	RP←((SP + 1),(SP))	POP	RP (2)	10
		POP	IR	14
Write to Stack	((SP – 1),(SP – 2))←RP	PUSH	RP (2)	11
		PUSH	IR	15
Swap Top of Stack	((SP + 1),(SP))⇄RP	EX	(SP),HL	19
		EX	(SP),IR	23
Arithmetic Operations				
Add Memory	A←A + M	ADD	A,(HL)	7
		ADD	A,(IR + D)	19
Add Memory with Carry	A←A + M + F[C]	ADC	A,(HL)	7
		ADC	A,(IR + D)	19
Subtract Memory	A←A – M	SUB	(HL)	7
		SUB	(IR + D)	19
Subtract Memory with Carry	A←A – (M + F[C])	SBC	(HL)	7
		SBC	(IR + D)	19
Decrement Memory	M←M – 1	DEC	(HL)	11
		DEC	(IR + D)	23
Increment Memory	M←M + 1	INC	(HL)	11
		INC	(IR + D)	23
Logical Operations				
AND Memory	A←A↑M	AND	(HL)	7
		AND	(IR + D)	19
OR Memory	A←A↓M	OR	(HL)	7
		OR	(IR + D)	19
Exclusive OR Memory	A←A≠M	XOR	(HL)	7
		XOR	(IR + D)	19
Test Bit X in Memory	F[Z]←~M[X]	BIT	X,(HL)	12
		BIT	X,(IR + D)	20
Set Bit X in Memory to 1	M[X]←1	SET	X,(HL)	15
		SET	X,(IR + D)	23
Reset Bit X in Memory to 0	M[X]←0	RES	X,(HL)	15
		RES	X,(IR + D)	23
Compare Memory	Set Flags	CP	(HL)	7
		CP	(IR + D)	19
Search Memory, Step-Wise	Set Flags	CPD		21/16
		CPI		21/16
Search Memory, Block-Wise	Set Flags	CPDR		21/16
		CPIR		21/16
Manipulative Operations				
Shift Memory Left, Arith.	F[C]←M←0	SLA	(HL)	15
		SLA	(IR + D)	23
Shift Memory Right, Arith.	M[7]→M→F[C]	SRA	(HL)	15
		SRA	(IR + D)	23
Shift Memory Right, Logical	0→M→F[C]	SRL	(HL)	15
		SRL	(IR + D)	23
Rotate Memory Left thru Carry	F[C]←M←F[C]	RL	(HL)	15
		RL	(IR + D)	23
Rotate Memory Left Circular	F[C]←M←M[7]	RLC	(HL)	15
		RLC	(IR + D)	23
Rotate Memory Right thru Carry	F[C]→M→F[C]	RR	(HL)	15
		RR	(IR + D)	23
Rotate Memory Right Circular	M[0]→M→F[C]	RRC	(HL)	15
		RRC	(IR + D)	23
Rotate BCD Left thru Memory		RLD		18
Rotate BCD Right thru Memory		RRD		18

Table 2. Z-80 instructions involving memory.

faster than a study of the texts.

Advisory No. 1

Decide on the type of instruction you need, such as add memory, increment a value, read an external device. Look it up in the appropriate table. Make a tentative selection that best matches your purpose. Look up details of the instruction in your reference sources. Make a different choice if necessary. As you develop sequences of instructions, you will firm up your specific allocation of registers, choice of addressing modes and use of flags. This may call for one or more iterations of the selection process. When you have done everything, you are ready to either assemble or hand-code your program.

Chopping Wood—A Worked Example

An example will clarify the entire process. We will write a program to copy a block of memory from one location to another and then return control to the TRS-80 Level II 16K BASIC system. All our address references and byte counts will be in hexadecimal.

Specifically, we want to copy 600 bytes of memory starting at location 4380 in the BASIC user's area to a location starting at 6380 in the protected user's machine-language area. The program is quite small, so we will hand-assemble it starting at location 7001. Once written, it is conveniently entered in a few seconds through the keyboard with the aid of a machine-language monitor, although it could be POKEd if necessary.

As a starter, we write the simple procedure in an improvised language using whatever phrases appear convenient:

Set source address to 4380
Set target address to 6380
Set counter to 600
Copy (read and write) bytes until counter is satisfied
Transfer control to BASIC at 1A19

We could just as well have used expressions such as GOTO 1A19 or CTR = CTR + 1 or

diagrams . . . anything as long as the procedure is made reasonably clear.

The key operation here is the copy. Since it is a memory-to-memory copy that we want, we look in Table 2 (memory instructions). There we find under "Data Transfer" two block copy instructions, LDDR and LDIR. Look them up in one of the references to learn that LDIR increments memory addresses as it copies, which is what we want. We also learn that this instruction uses three register pairs, HL, DE and BC.

In these it expects to find, respectively, the source memory address, the target memory address and the number of bytes to be copied. The Z-80 automatically increments the memory addresses and decrements the byte counter until it goes to zero.

Our next task is to load or store the required values in these three register pairs. Looking up Table 1 (register instructions), we find again under "Data Transfer" three "store constant" instructions, one of which concerns register pairs, namely, LD RP,NN. These are to the point, taking the form LD HL 4380, for example.

The final item to be looked up is a suitable transfer of control instruction in Table 3. Here a simple "jump to address" instruction, JP AD, is appropriate. We can now write our program, with comments:

```
LD    HL,4380   : Point to first source
                  location
LD    DE,6380   : Point to first target
                  location
LD    BC,600    : Set counter for bytes to
                  be copied
LDIR            : Copy block of memory
JP    1A19      : Transfer control to
                  BASIC
```

Normally, we would use an assembler, but this simple program is easily hand-assembled. You should consider hand-assembly of your first machine-language program. It will give you a direct feel for what is going on. To hand-assemble, look up each mnemonic in a Z-80 cross-assembler table such as is found in the *TRS-80 Editor/Assembler, Operation and Reference Manual*. This will give you the corresponding machine language or object code, which you write on the left of the corresponding mnemonic. Thus:

```
21 80 43  LD    HL,4380  : "from" address
11 80 63  LD    DE,6380  : "to" address
01 00 06  LD    BC,600   : number of bytes
ED B0     LDIR           : copy memory
C3 19 1A  JP    1A19     : go to BASIC
```

Each two-digit hex number on the left represents a byte of object code. But wait, what has happened to the addresses! They are reversed with the low-order byte going in before the high-order byte. This is simply one of the machine conventions, which abound in machine language. If you are new to machine language, you will soon get used to it.

All that remains now is to enter the 14 bytes of code into memory starting at location 7001, for our example. A machine-language monitor such as T-BUG for the TRS-80 is a convenient way to do this. If you use this monitor, you will recognize that our simple example copies T-BUG itself into upper memory.

To actually perform the copy, you simply use the monitor's jump command to transfer control to location 7001.

If you wish to verify the copy, simply change the last instruction to JP 43A0 or C3 A0 43. This will return you to T-BUG instead of BASIC. You can now examine the copy and then use the jump command of the monitor to restore control to BASIC. The copy of T-BUG will not run correctly, since many of its addresses will reflect inconsistent references (4000 memory block versus 6000 block).

Correcting the addresses in the copy is known as relocation, which is a subject in itself. The point is to recognize that machines deal with absolute addresses only. If your application requires relocatability, you may achieve this with the aid of an assembler and the choice of certain addressing modes available in the Z-80.

Although the above example is trivial, it has illustrated the principle of using the tables as a starting point to find the exact instructions you need and then reading up on them as necessary, avoiding all unnecessary details in the process. If your next problem requires use of an output port, for example, you will be directed to the correct instructions by the same process and proceed to learn just what you need to know.

Having taken the trouble to read this article, by all means try writing a simple program . . . anything, even if it is only to add one and one! You will have broken through a barrier that, in the case of the Z-80, can appear quite formidable. You will soon come to appreciate the versatility and power of your Z-80 processor.

The LDIR instruction we just employed is a good example; it performs two increments, one decrement and an exit test, all automatically within a loop that

Function	Definition	Mnemonic	Operands (opd)	Clock Cycles
Input/Output				
Read Device into Register	R←(opd)	IN	A,N (4)	10
		IN	R,(C) (3)	11
Write Register to Device	(opd)←R	OUT	N,A (4)	11
		OUT	(C).R (3)	12
Read Device into Memory, Step-wise		IND		15
		INI		15
Read Device into Memory, Block-wise		INDR		20/15
		INDR		20/15
Write Memory to Device, Step-wise		OUTD		15
		OUTI		15
Write Memory to Device, Block-wise		OTDR		20/15
		OTIR		20/15
Interrupts				
Disable Interrupts		DI		4
Enable Interrupts		EI		4
Execute Device's instruction or interrupt		IM	X	8
Transfer of Control				
Call Subroutine (transfer control to)		CALL	LBL(or AD)	17
Call Subroutine if Condition of flag is met		CALL	COND,LBL	10/17
Call Subroutine at Page Zero Addresses		RST	XX	11
Return from Subroutine		RET		10
Return from Subroutine if Condition is met		RET	COND	5/11
Return from Interrupt (with Priority)		RETI		14
Return from Non-Maskable Interrupt		RETN		14
Jump (go to)		JP	(HL)	4
		JP	(IR)	8
Jump to Label (or Address AD)		JP	LBL	10
Jump to Label, on Condition		JP	COND,LBL	10
Jump to Displacement from Program Counter		JR	DISP	12
		JR	COND,DISP (9)	7/12
Jump if Counter is Non-Zero (loop)		DJNZ	DISP	8/13
Miscellaneous				
Pause for 4 Clock Cycles		NOP		4
Halt Execution (Until Reset or Interrupt)		HALT		4

Table 3. Z-80 instructions for input/output and execution control.

Unless Otherwise Noted:

AD	= Explicit address
COND	= Condition, i.e., the state of a flag
D	= Displacement, in context IR + D
DISP	= Displacement, range of − 126 to + 129 bytes or memory locations
IR	= Index registers IX or IY
LBL	= Label or symbolic address
M	= Memory
N	= One-byte data value, sometimes an I/O port address
NN	= Two-byte data value, often an address
R	= Registers A,B,C,D,E,H or L, i.e., all 8-bit registers except F,I and R
RP	= Register pairs BC, DE or HL, or 16-bit registers IX, IY or SP
X	= Decimal digit in range 0–7 for bits or 0–2 for interrupt modes (IM X)
XX	= One of eight locations 00, 08, 10, 18, 20, 28, 30, 38 Hex
opd	= Operand
( )	= Contents of enclosed item; e.g., (HL) signifies location pointed to by content of HL register pair, i.e., an indirect memory reference
[ ]	= Subscript, e.g., F[C] = Carry flag bit in flag register F
←	= Store in or shift left
→	= Go to or shift right
~	= Negation of (logical)
↑,↓	= And, or (logical)

Notes

(1)	= Register pairs BC, DE or HL only
(2)	= Register pairs AF, BC, DE or HL only
(3)	= R includes F register; register C contains device port address
(4)	= N is port or device address
(5)	= BC, DE HL or SP only
(6)	= R is interrupt vector register (I) or refresh register (R) only
(7)	= BC, DE, IX, IY or SP only
(8)	= Accent (') denotes alternate register set; for RP' all three pairs are swapped (exchanged)
(9)	= Flags C, NC, Z, NZ only

Table 4. Explanations and notes for Tables 1, 2 and 3.

is also set up and processed automatically. Pretty good for just two bytes! There's a lot more of that power sitting in your machine just waiting to be used.

The Way out of the Forest

We pointed out in part 1 that the purpose of this article was not to teach Z-80 programming. Clearly, you will be teaching yourself. What we have done is provide an approach that prevents hang-ups . . . hang-ups on obscure points, on too much information, on too many choices, on too much detail.

The key is to identify your needs in stages with increasing specificity as you narrow down the choices. Only then do you go into detail on a specific item. This is really an informal type of top-down program design and implementation aimed at learning while doing.

Aside from encouraging the use of this approach and your own efforts, the three tables offered here are the only other substantial element in this scheme. If you intend to master Z-80 assembly-language programming, you may want to develop your own form of such tables. It represents a fairly heavy effort, but the exercise will reveal a great deal about the Z-80.

Advisory Finale

Don't get hung up on detail, choices, obscurities. Be resourceful (use parallel reading). Identify your problem. Narrow it down to specifics. Then, and only then, get as detailed as the solution requires.

After all, the riches of the Z-80 ought to be a comfort, not an embarrassment. ■

References

1. William Barden, Jr., *The Z-80 Microcomputer Handbook*, Howard W. Sams & Co., Inc., Indianapolis IN, 1978.

2. *Mostek Z-80 Technical Manual*, MK 3880 CPU, MP 1031, and *Mostek Z-80 Programming Manual*, MR 78515, Mostek Corp., Carrollton TX, 1977.

3. Adam Osborne, et al, *Z-80 Programming for Logic Design*, Osborne & Associates, Berkeley CA, 1978.

4. *TRS-80 Editor/Assembler, Operation and Reference Manual*, Catalog No. 26-2002, Radio Shack, Tandy Corp., Ft. Worth TX, 1978.

5. N. Wadsworth, *Z-80 Instruction Handbook*, Scelbi Publications, Milford CT, 1978.

Teamwork!

When you add the Selectra-Print System 9710 to your computer or terminal, you've got the winning team working for you!

Selectra-Print gives you letter-quality hard copy for your system plus the versatility and efficiency of IBM's world-famous Selectric II.™ At a fraction of the price you would expect to pay for a conventional printer alone.

Just plug it in to any computer or terminal. Selectra-Print is compatible with all serial and parallel ports. No program modification or additional interface is required. If you own a TRS-80,™ there's a special Selectra-Print to meet your needs.

Using standard ASCII code, Selectra-Print brings all available keyboard characters under computer control. You can now mail, copy or file your computer or terminal outputs. They are even camera-ready for your local offset printer.

In your office, you can use Selectra-Print to take over the typing chores. Just program your computer for the desired output, and Selectra-Print delivers forms, correspondence and other documents, letter-perfect. Addresses, dates and other variables can be simply typed either manually or by your computer or terminal.

Your Selectra-Print can be used as a high quality office typewriter, too. Choose either an easy-reading 10 characters per inch or a space-saving 12 characters per inch. Or you may order your System 9710 with dual pitch, which incorporates both. For maximum versatility, correction is available as an option for all systems.

Selectra-Print comes complete with an IBM warranty on the typewriter. Micro Computer Devices warrants the rest. If, after the warranty period expires, you would like an annual service contract, you may purchase one for a nominal charge. So why pay more for less? If you own a computer or terminal, why not really put it to work? With Selectra-Print and IBM, you've got the hard copy team!

Standard Features

- IBM standard warranty on Selectric II™ typewriter
- MCD standard warranty on Selectra-Print
- Finest printed copy available
- ASCII typing element
- 130 characters per line at 10 per inch, or 155 characters at 12 per inch
- Full upper and lower case alphanumeric characters
- Parallel or serial interface
- Switch-selectable rates to 9.6 kbaud (serial port only)
- Tab command
- Backspace
- Index
- Bell
- Computer cable set
- All necessary software in PROM
- Choice of IBM colors

Optional Features

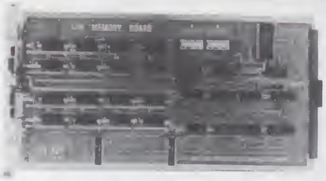
- Dual pitch
- Correction
- Noise reduction
- RS-232 serial interface
- Pin feed
- Tractor feed
- All available typing elements
- Annual IBM service contract for Selectric II™
- Annual MCD service contract for Selectra-Print

MICRO COMPUTER DEVICES, INC. 24

3156 East La Palma Avenue, Suite H, Anaheim, CA 92806 · (714) 630-8206

Reader Service—see page 257

HEATH ® COMPATIBLE



DG-32D 32K RAM FEATURES:

- Plugs into Heath® H8 Computer.
- Ready to use. Fully assembled, tested & burned in.
- Operates with existing Heath® memory.
- Protected memory Output Buffers in the event of Address error.
- Utilizes popular 4116 RAM devices.
- Memory Address DIP switch changeable.
- Arranged as 4 Independent 8K Blocks.
- Low Power Consumption: Less than 6 watts, typical.
- Transparent Refresh.
- One year guarantee.
- Compatible with all current H8 peripherals.

D•G Electronic Developments Co. brings you a totally compatible, fully assembled and tested 32K RAM for Heath® H8 computers. The DG-32D has less than 6 watts power consumption. This allows you to add a full 32K bytes of Random Access Memory without taxing or replacing your computer's power supply. Engineered to plug-in and run without any user modifications, the DG-32D can be used with or without existing H8 RAM without modification. Protection of the memory output buffers is provided in the event of assigning two blocks to the same address space. The DG-32D is the ideal answer to expansion of the Heath® H8 computer. . . Low power consumption, low price, high capacity, total engineering and exacting production methods.

Heath® and H8 are registered trademarks of the Heath Corporation, Benton Harbor, Michigan.

BURNED IN 16K MEMORY CHIP SET
H88-H89 .... **$89.00**

ORDERING INFORMATION: DG-32D RAM available only from DG Electronic Developments Co., P.O. Box 1124, 1827 South Armstrong, Denison, Texas 75020. Check, money-order, VISA or Master Charge. Phone orders accepted on charge orders. NO COD's. Foreign orders add 30%. Texas residents add 5%. For VISA or Master Charge orders call 214-465-7805. $479.00 freight prepaid. Allow three weeks for personal checks to clear banks.

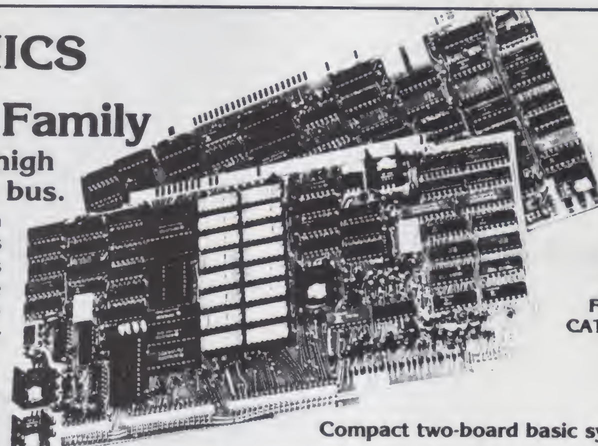
D·G ELECTRONIC DEVELOPMENTS CO. ✓159

32K RAM $479.00

Complete COLOR GRAPHICS With The CAT-100 Family

The original 256-color imaging system with high resolution video frame grabber for the S-100 bus.

Its digitizer captures a video frame in 1/60th of a second and stores it in its on-board image memory, with software selectable resolutions ranging from 256 to 1280 pixels per TV line. The image memory is accessible through a 2K window in the S-100 address space. The compact two-board basic system digitizes and displays 480x512 single bit pixels or 240x256 four bit pixels. The video generator displays 16 gray levels or 16 colors on standard B&W, NTSC or RGB color TV monitors.

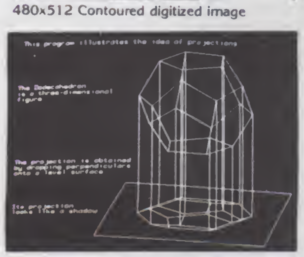


FREE CATALOG

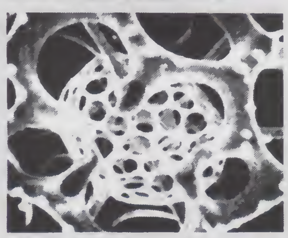
Compact two-board basic system

480x512 Contoured digitized image

240x256 Digitized image, 16 levels



480x512 Computer-generated



240x256 Digitized image, 16 levels

Features:

- Input capability from RS-170 and other TV signals
- Variety of synchronization choices
- 2 selectable video A/D conversion circuits
- Choice of 1, 2 or 4 bits per pixel
- 32K-byte image memory on the basic unit
- Expandable to 256K bytes with 24-bit color map
- Lightpen input
- Photographic trigger control input
- Character generator
- Software selectable system parameters
- Interfaces for LSI/11, TRS-80 and other processors

Accessories:

- NTSC and RGB Sony Trinitron color monitor
- High sensitivity, high resolution lightpen
- Hard copy electrostatic printer/plotter with interface
- Large variety of software packages

ROBOTICS·AUTOMATED COUNTING AND MEASURING·QUALITY CONTROL MONITORING·PATTERN RECOGNITION·IMAGE PROCESSING·MEDICAL AND SCIENTIFIC RESEARCH

DIGITAL VIDEO SYSTEMS
441 California Ave.,
Palo Alto, CA 94306
415/494-6088
Dept. 36

✓73

C. Kevin McCabe
115 South LaSalle St.
Suite 3300
Chicago IL 60603

Conjure up a GET Command for Sorcerer

Input is a pain in the return key.

The Exidy Sorcerer has many of the abilities—dense graphics, 128 user-defined characters and a relatively fast version of Microsoft BASIC, to name but a few—needed for arcade-like game programs. A quick check of the label on the BASIC ROM PAC, though, may leave a question in the user's mind: Who got the GET command?

An obstacle-avoidance program or a fancy Star Trek routine should allow a player to specify movements with, for example, the four cursor directional keys in the numeric keypad. Rather than an INPUT statement, which requires an on-screen prompt and use of the RETURN key, there should be some means of informing a BASIC program that any specific key has been pressed.

Normally, such inputs would be received using a GET command, as on the Commodore PET. In fact, GET is listed on the ROM PAC label. But somewhere between the print shop and the ROM programmer, the GET command disappeared from the Sorcerer's bag of tricks. Are we forever bound to clumsy INPUT statements, or is there some way to get a GET with what we've got?

The Magic Elixir

The USR function is just the hocus-pocus the Sorcerer ordered. USR causes BASIC to execute a machine-language subroutine, then return to the BASIC program. The phrase "machine language" may scare a few confirmed BASIC hackers, but it needn't. Remember that machine-language commands to load registers, set flags and do all manner of other esoteric wizardry are nothing more than sequential bytes stored in memory. Those bytes, in turn, can be POKEd into place by a BASIC program, with no need for an assembler.

The BASIC subroutine in Listing 1 sets up a machine-language replacement for GET in the first eight memory locations (leaving 0000H blank to hold results). The values POKEd into addresses 260 (0104H) and 261 (0105H) direct the machine to this particular routine whenever USR is called by an X = USR(0) statement. If data is then available from the keyboard, the routine will load a value between 1 and 255 into memory location zero; if no key is pressed, the stored value will be zero.

For alphanumeric and special characters and control codes, the stored value is simply the ASCII value of the pressed key (taking into account any use of SHIFT or SHIFT LOCK). If the GRAPHIC key is also pressed at the same time, a value corresponding to the standard or user-defined graphics character will be stored.

Listing 2 illustrates a short BASIC program using the GET replacement routine. Line 200 repeatedly calls the routine, using the dummy argument and scratch variable shown, until a key is pressed. Any nonzero value loaded into memory location zero by USR will be printed, then the keyboard search will resume until another input is found or CONTROL-C halts the program.

With one or more IF or ON statements, a BASIC program can select alternative actions based on user input. The program in Listing 3 moves a graphic character within the confines of the screen. The user makes a single-space move by pressing one of four direction keys. Line 100 specifies the ASCII comparison values used to determine the direction of movement.

In this instance, the 2, 4, 6 and 8 keys (ASCII 50, 52, 54 and 56) are checked—a glance at those keytops in the numeric pad shows why those values were selected. While the program could just as easily check for the cursor-related control codes (ASCII 26, 1, 19 and 23) generated by those same keys, the input wouldn't be recognized unless a SHIFT key was also depressed.

The Formula

How does the Sorcerer accomplish this magic? It's done with a combination of Monitor and Z-80 commands, along with BASIC's USR function. Table 1 shows the various memory locations and values used by the GET replacement routine.

On power-up or RESET, Sorcerer's BASIC interpreter automatically loads a value of C3H into address 0103H. As shown, C3H is the hex code for the Z-80 command JP mn, that is, jump to (and continue execution with) address mn.

In machine-language routines, addresses that require 16 bits are specified by the two consecutive bytes immediately following a jump, call or other address-referenced command. The first byte (n) is the low-order, or least significant, address byte; the second byte (m) is the high-order portion of the address.

Sorcerer's BASIC calls the subroutine at address 0103H whenever a statement of the form X = USR(0) is encountered. Since BASIC has already stored the JP mn command in that location, the contents of the next two addresses simply specify the starting address mn of the desired machine-language subroutine.

Although such subroutines can be stored anywhere in RAM, addresses 0000H through 00FFH work best; this portion of memory is never used by either BASIC or the Monitor for any other purpose. The machine-language routine set up by Listing 1's code begins at address 0001H, so values of 01H and 00H are POKEd into locations 0104H and 0105H, respectively (remember the reversed order).

```
6ØØØØ REM -- Set up USR subroutine
6ØØ1Ø RESTORE:  DATA 2Ø5, 9, 224, 5Ø, Ø, Ø, 2Ø1
6ØØ2Ø FOR ADDRESS = 1 TO 7: READ MLANG
6ØØ3Ø POKE ADDRESS, MLANG: NEXT ADDRESS
6ØØ4Ø REM -- Specify starting address for subroutine
6ØØ5Ø POKE 26Ø, 1: POKE 261, Ø: RETURN
```

Listing 1.

```
100 GOSUB 60000: REM--add lines 60000-50 in Listing 1
200 X = USR(0): IF PEEK(0) = 0 THEN 200
300 PRINT PEEK(0): GOTO 200
```

Listing 2.

Execution of the selected machine-language subroutine begins with the command found at the specified starting address. The command for Listing 1's subroutine is the Z-80's CALL mn, with the subroutine address mn being E009H as specified by the next two (reversed order) bytes in memory. Address E009H isn't in RAM at all; in fact, it's an "entry point" into the Monitor ROM routines.

Entry here causes any available single-byte input to be loaded from the current input device (normally the keyboard) into the Z-80's A register. If no input is found, a zero value is loaded instead. After the loading operation, control returns to the command following the CALL to the Monitor routine—in this case, the command at address 0004H.

This next command is LD(mn),A, which causes the Z-80 to load the contents of its A register into a designated memory address mn (again specified by the next two bytes). In Listing 1's routine, mn is address 0000H.

The final command of the USR routine, located at the next available address (0007H), is the RET instruction, which returns control to the BASIC program. At this point, PEEK(0) will reveal the input value or equal zero if no input is found.

Modifications

A few interesting variations are possible. For example, you can load the program shown in Listing 3, then enter the Monitor and input an SE I = P command. On return to BASIC (by the Monitor command PP), the current input device will be the built-in parallel input port FFH. Any device connected to input port FFH, such as a joystick-controlled analog-to-digital converter, will now determine the values found by the USR routine. Monitor command SE I = K returns the keyboard to its normal status as the current input device.

Notice that any number of USR subroutines can be contained in a program. By POKEing the desired starting address (in decimal) into locations 0104H and 0105H, successive calls to USR may execute different machine-language routines. Remember to use the RET command as the last byte of each routine, so that control returns to the BASIC calling program.

As shown, USR isn't particularly difficult to use, even for those programmers unfamiliar with Z-80 op codes. Most of the texts on Z-80 programmming will have tables of commands and their associated hex or binary values. Monitor and USR routines are briefly discussed in Exidy's *Sorcerer Technical Manual*. With this data, machine-language routines needn't intimidate BASIC programmers any longer. ■

Address Decimal	Address Hex	Stored Value Decimal	Stored Value Hex	Remarks
0	0000H	ASCII input or zero		Holds results of input search routine.
1	0001H	205	CDH	Z-80 CALL mn command
2	0002H	9	09H	Low-order address byte of called routine(n)
3	0003H	224	E0H	High-order address byte of called routine (m).
4	0004H	50	32H	Z-80 LD(mn), A command Loads mn with one-byte value in A register
5	0005H	0	00H	Low-order address byte for load (n).
6	0006H	0	00H	High-order address byte for load (m)
7	0007H	201	C9H	Z-80 RET, returns control to BASIC
259	0103H	195	C3H	Z-80 JP mn command, causes jump to routine at mn
260	0104H	1	01H	Low-order address byte of desired USR routine
261	0105H	0	00H	High-order address byte of desired USR routine
8183	E009H	Monitor RAM		Monitor input RECEVE entry point. Loads one byte, if available, from current input device into A register

Table 1. Memory locations and values used by the GET replacement routine.

```
100 U = 56: D = 50: L = 52: R = 54
110 DOTAT = -2985: PRINT CHR$(12);
120 GOSUB 60000: REM--add lines 60000-50 in Listing 1
200 POKE DOTAT, 132
210 X = USR(0): KEY = PEEK(0): IF KEY = 0 THEN 210
220 POKE DOTAT, 32
300 IF KEY <> L THEN 400
310 IF DOTAT / 64 = INT(DOTAT / 64) THEN 200
320 DOTAT = DOTAT - 1: GOTO 200
400 IF KEY <> R THEN 500
410 IF (DOTAT + 1) / 64 = INT((DOTAT + 1) / 64) THEN 200
420 DOTAT = DOTAT + 1: GOTO 200
500 IF KEY <> U THEN 600
510 IF DOTAT - 64 < -3968 THEN 200
520 DOTAT = DOTAT - 64: GOTO 200
600 IF KEY <> D THEN 200
610 IF DOTAT + 64 > -2049 THEN 200
620 DOTAT = DOTAT + 64: GOTO 200
```

Listing 3.

Reader Service—see page 257

88

Interesting

Solve problems arising from use of textbook formulas to calculate compound interest.

Sid Owen
246 Walter Hays Dr.
Palo Alto CA 94303

Frustrating! Your pocket calculator comes close to the interest posted in your savings-account book, but your expensive computer indicates a larger error.

The textbook formula for compound interest frequently yields significant errors when used with home computers. A simple solution, along with some comments on bank practices, is discussed below. The programs are written in North Star BASIC with notes about changes required for other BASIC languages. Some algebra is included, but only for explanation of the derivations—the routines are just as useful without the mathematical background.

The Accuracy Problem

The basic formula for calculations with compound interest is:

$P = A(1 + I)^N$

where P = present value; A = amount on deposit; I = interest rate; and N = number of times compounded. This simplest version of the formula is used for annual compounding.

Since bank-account interest rates are normally quoted as the annual simple rate, that is, the noncompounded rate, the interest term must be divided by M, the number of times interest is compounded per year. The resulting I/M is the interest earned for one compounding period. The basic formula then becomes:

$P = A(1 + I/M)^N$

This is a simple formula that is easily programmed. If you want accurate answers, *don't use it*.

The I/M term in the parentheses may be very small, resulting in several zeros between the decimal point and the significant digits. The term by itself is no problem for floating-decimal languages. However, when the 1 is added to this small value, a language with eight-digit precision rounds off the sum to eight digits, saving the zeros, but dropping the excess digits from the interest term. This loss of interest accuracy significantly affects the final answer.

For example, the daily interest rate for a 5.25 percent account is 0.00014383562. The (1 + I/M) term is rounded to 1.0001438, which would result in an annual interest error of about \$.14 on a \$10,000 account (see Run 1). Add in the effects of several bank practices such as described below, and your computer is not much help in predicting or verifying the bank's calculations! In addition, some computer algorithms for raising variables to powers have been known to introduce small errors.

Accurate Interest

The compound-interest equation can be rapidly and accurately solved by the use of the binomial expansion.

See the binomial expansion in Equation 1. The compounded interest rate is the sum of all the terms except the 1, so that the full precision is available for the interest. In theory, there are as many terms in the binomial expansion as there are periods or days of compounding, but in practice, the terms become negligibly small after five or so. Also, each term may be derived from the previous term so that a loop routine can be used in your program. This routine is illustrated in lines 390 through 500 of Listing 1.

In each pass, the loop computes a term from the previous one, saves the new term for the next round and also adds it to the series total. The result is compared with the previous total. If the term was not significant enough to raise the total (within the precision of your language), then the best interest within the limits of your system has been computed and the program will exit the loop.

This routine does not need changes for languages of different precisions. Interest

```
RUN

Type output device number: 0
Type max. desired decimal places printed (0 to 8): 8

                    INTEREST

Annual nominal interest (%)            5.25
Compounded how many times annually?    365
Number of times compounded?            365
Amount on deposit?                  $  10000

                    By Binomial                 By
                     Expansion            ((1+(I/M))^N) -1
---------------------------------------------------------------
Effective rate  =    5.3898583 %              5.38849 %

Interest earned =     $538.99                 $538.85
---------------------------------------------------------------

READY
```

Run 1. Shows the error when the conventional routine is used.

Listing 1. Program prints interest by conventional and by binomial routines.

```
80       REM 'BinomI' BINOMIAL INTEREST, by S. Owen, 3/13/79
90  REM
100 INPUT "Type output device number: ",W
110 !"Type max. desired decimal places printed (0 to 8): ",
120 INPUT "",D
130 D=10^(D+2)\!
140 !#W,TAB(20),"INTEREST"\!#W
150 INPUT #W, "Annual nominal interest (%)           ",I9
160 INPUT #W, "Compounded how many times annually?  ",M
170 IF M THEN 180\!#W,"Type 1 for simple interest. "\GOTO 160
180 I=I9/M*.01 \     REM I=interest for one period, e.g. one day
190 M1=M-360=0\      REM Flag 360 day years
200 INPUT #W, "Number of times compounded?           ",N\N1=N
210 IF N THEN 220\!#W,"Type 1 for simple interest. "\GOTO 200
220 IF M1=0 THEN 250
230 INPUT #W, "Credited quarterly?                   ",Q$
240 IF Q$(1,1)="Y" THEN Q=1
250 INPUT #W, "Amount on deposit?                 $  ",A\A3=A
260 !#W
270 !#W,TAB(20),"  By Binomial                    By"
280 !#W,TAB(20),"   Expansion             ((1+(I/M))^N) -1"
290 FOR J=1 TO 63\!#W,"-",\NEXT J\!#W
300 GOSUB 390
310 GOSUB 530
320 IF Q*M1 THEN GOSUB 650 ELSE GOSUB 590
330 FOR J=1 TO 63\!#W,"-",\NEXT J\!#W
340 !#W,%#,
350 !#W
360 END
370 REM
380       REM calculation subroutine
390 E=N*I\T1=E\      REM We're computing 'E'ffective interest
400 FOR J=2 TO N
410 E9=E\            REM Save last interest value
```

calculated with an eight-digit BASIC will be comparable to your bank's calculations, agreeing exactly with some banks and within a penny or so with others.

Continuously Compounded

Government regulations set the maximum annual simple (i.e., nominal) interest rate for banks, trust and loan companies and savings and loan companies (all called "banks" in this article). The banks, however, can compound as often as they want for competitive reasons.

Computers made it simple for banks to go to daily compounding, and some banks even took the last possible step of continuous compounding, which is also simple to do with computers. Each increase in the number of times that interest is compounded results in a higher effective annual rate, but each increase has a smaller effect. The mathematical limit is "continuously compounded" interest, and present value can be computed with the simple formula:

$P = Ae^{I}$

where e = 2.7182118 (the base of natural logarithms) and I = nominal annual interest rate.

The computed e-to-the-I-term results in a number that is actually the sum of 1 plus the effective interest rate. The formula therefore says, "Present value equals A (times the 1) plus A times effective interest rate." The effective interest rate itself can be isolated by subtracting 1 from e-to-the-I, as implemented in line 410 of Listing 2.

$$(1+I)^N = 1 + NI + \frac{N(N-1)}{2!} I^2 + \frac{N(N-1)(N-2)}{3!} I^3 + \ldots + I^N$$

Equation 1. Binomial expansion.

Notice in Run 2 that there are only six or seven digits for continuous interest. Before subtraction, the eight digits of precision include the one and maybe a zero. There are only six or seven digits left for interest after subtraction.

Continuous compounding yields interest rates that are slightly higher than daily compounding, but trying to explain continuous interest to customers (and tellers) must be difficult for the banks. Continuous interest appears to be a dying fad.

The 360-Day Year

Most banks and many other businesses define a year as 360 days with 12 months of 30 days each. The reason is obviously not the dime or so that they save each year for every million dollars you have in your 5 percent account.

```
420 T2=T1*I*(N+1-J)/J\           REM Compute next term
430 E=E+T2\             REM Add new term
440 T1=T2\              REM Save new term
450 IF E=E9 THEN EXIT 470\       REM Quit if no more progress
460 NEXT
470 E1=100*INT(E*D+.5)/D\        REM Round-off
480 E3=-1+(1+I)^N\  REM Compute textbook formula
490 E4=100*INT(E3*D+.5)/D\       REM Round-off
500 RETURN
510 REM
520       REM Print rate subroutine
530 !#W,%Z12F8, "Effective rate  =      ",E1,"% ",
540 !#W,TAB(48),E4," %"
550 !#W
560 RETURN
570 REM
580       REM Print earnings subroutine
590 R=A*E\R=INT(100*(R+.005))/100\A=A+R\           REM Round-off
600 !#W,"Interest earned =  ",%#$C12F2,R,
610 R3=A3*E3\R3=INT(100*(R3+.005))/100\A3=A3+R3\ REM Round-off
620 !#W,TAB(45),R3
630 RETURN
640 REM
650       REM 90 day quarterly credit subroutine
660 N=90\GOSUB 390\              REM Calc. 90 day rate
670 IF N1<90 THEN 710
680 GOSUB 590\                   REM Print
690 T=T+R\T3=T3+R3\N1=N1-90\     REM Accum. totals
700 GOTO 670
710 IF N1=0 THEN 770
720 N=N1\GOSUB 390\              REM Calc. short term rate
730 GOSUB 590\                   REM Print
740 T=T+R\T3=T3+R3\              REM Accum.
750 REM
760        REM Print total routine
770 !#W\!#W,"   TOTAL          =  ",T,
780 !#W,TAB(45),T3
790 RETURN
```

The real advantages (for both you and the bank) are that all months are the same; there are no leap years, and the months and quarters are exactly 30 and 90 days each. From January 1 to March 3 is two months and two days, or 62 days. July 1 to September 3 is also 62 days. Tellers make fewer errors and computer programs are much simpler.

"Credited Quarterly"

A savings account compounded daily usually includes fine print that says something like "earnings distributed quarterly," or "credited quarterly." This means that the interest is being accumulated for the account daily, but it will be posted in the savings book quarterly, and for good reason.

When posted (i.e., distributed or credited), the interest and principal plus interest are rounded off, eliminating fractions of a cent. Interest then starts compounding on the rounded-off value for the next quarter. However, the account has earned the unposted interest (since the last posting) and it would, for instance, be rounded off, posted and paid if the account were closed.

Without the restriction to quarterly posting, a single day's interest on small accounts could round off to zero and result in no annual interest if the bank posted interest when you presented your savings book each day.

Interest is credited quarterly, whether or not it is actually posted in your savings book. Rounding off at these quarterly postings may result in a difference of a penny or two per year, plus or minus, when compared to interest that is calculated using annual rate formulas or rates from tables. The amount of difference depends upon the exact balance on deposit. Listing 1 will take posting into account if the "Credited quarterly?" question is answered "Yes," as shown in Run 3.

Other Banking Variables

"In by the 10th. Interest from the 1st." This is another advertising technique to get more customers. For your calculations, it is easily handled by adding the extra days of compounding.

Balance averaging is frequently used to reduce the bank's computer load. With an active account, the interest computation is based on the average balance (or else a weighted average balance) over some period. If your account is active, then checking interest calculations will require knowledge of the specific methods used by your bank.

T-Bill Accounts

Lucky you! You just received a $10,000 birthday check from Aunt Tillie, and it is Wednesday. You have decided to open one of those new high-interest "Money Market" or "T-Bill" certificate accounts, which have nominal interest based on the government Treasury Bill interest for the week. Each Wednesday you can phone and find out the interest for the week starting Thursday, continuing through next Wednesday.

If tomorrow's rate is higher, compute how much more the higher rate would earn and then compute how much you lose by waiting a day without interest. The difference can be quite a few dollars.

Now you can make a logical choice—unless you want to guess about reinvestment rates at maturity. Also check your bank's fine print about interest for the week after maturity. If the high rate continues through that week, you can make another choice on the first Wednesday six months from now.

These certificates mature in 182 calendar days. Interest calculation practices vary.

```
80  REM `ITABLE', Binomial interest table, by S. Owen, 3/13/79
90  REM
100 DIM L(20)
110 M(2)=4\M(3)=12\M(4)=360\M(5)=365
120 INPUT "Type output device number: ",W
130 ! "Type max. desired decimal places in table (0 to 8): ",
140 INPUT "", D\D=10^(D+2)\!
150 !"Type list of simple interest rates, expressed as percent,"
160 !"ending list by typing digit 0:"
170 FOR J=1 TO 20
180 INPUT1 " ",I
190 IF I=0 THEN EXIT 230
200 !",",
210 L(J)=I/100
220 NEXT J
230 !
240 !#W
250 !#W,TAB(20),"EFFECTIVE INTEREST"
260 !#W
270 !#W," SIMPLE  QUARTERLY   MONTHLY       360 ",
280 !#W,"          365      CONTIN-"
290 !#W,"INTEREST   COMP.        COMP.      DAY YR.",
300 !#W,"     DAY YR.     UOUS"
310 !#W,%#Z11F8
320 FOR K=1 TO J-1
330 !#W,%Z7F4,L(K)*100," ",\        REM Print simple rates
340         REM Print compound rates
350 FOR P=2 TO 5
360 N=M(P)\I=L(K)/N
370 GOSUB 460
380 !#W,E1,
390 NEXT P
400         REM Calc. & print continuous rate
410 !#W,100*INT(D*(-1+EXP(L(K))))/D
420 NEXT K
430 !#W\!#W
440 END
450         REM calculation subroutine
460 E=N*I\T1=E
470 FOR R=2 TO N
480 E9=E
490 T2=T1*I*(N+1-R)/R
500 E=E+T2
510 T1=T2
520 IF E=E9 THEN EXIT 540
530 NEXT R
540 E1=100*INT(D*E)/D
550 RETURN
```

Listing 2. Program prints interest table for rates that you enter.

Some banks compute for 182 days of a 365-day year (but simultaneously use a 360-day year for other accounts!), while other banks compute for 180 days of 360, but pay only after 182 calendar days.

In the spring of 1979 the government prohibited interest higher than the actual Treasury Bill discount rate and also prohibited compounding during the 182-day period. It authorized the following payment formula (for savings and loan institutions):

(182/360)(earnings rate)(principal) = earnings

The government also authorized the following formula for advertising purposes:

$$(1 + (182/360)(\text{earnings rate}))^{(365/182)} - 1 = \text{Effective rate}$$

The rule changes have been frequent, so check with your bank before investing your gift money from Aunt Tille.

Comparative Interest-rate Table Program

Listing 2 shows a program using the above routines to print a comparative interest-rate table as shown in Run 2. To use it, type in the nominal interest rates that you want to compare, and the program will print the table. The format is deliberately crowded in order to print the table on a 64-character monitor screen. The questions and inputs appear on device number zero, but only the table is printed on the output device if it is other than zero.

Interest Shopping

Moving money from a 5 percent savings account to a 7.5 percent certificate account is only an increase of about 2.5 percent, right? Wrong! It is more than a 51 percent increase in profit! Just moving from 5 percent to 5.25 percent is an increase of more than 5 percent return on your investment. What businessman wouldn't jump at a chance of earning an extra 5 percent? Shop around, read the fine print and use these programs to get the most out of your savings. It pays—in cash!

Other BASICs

The listings may appear strange if you don't use North Star BASIC. Here is how to convert them to your BASIC:

1. Input and output devices—monitors, keyboards, printers, etc.—are identifed by a number (see Listing 1, line 100). Either substitute your language's method for identification or delete line 100 and "#W" at each occurrence.
2. ! means PRINT; !#W means PRINT on device number W. Either substitute PRINT for the !, or substitute your language's shorthand for PRINT.
3. In some languages, substitute [and] for (and), respectively.
4. A backslash (\) separates two independent statements with the same line number. The \ is equivalent to : in some languages. Otherwise, end any line when you get to a \ and continue with a new line number inserted before the next number in the printed listings.
5. Print format is defined by statements such as %Z12F8 in line 530, Listing 1. % means that format definition follows. Replace with your appropriate format definitions.
6. INPUT1 suppresses carriage return after the input. Forget this nicety and substitute INPUT.
7. Substitute GOTO for EXIT. In these simple programs you won't notice the difference. ■

```
RUN

Type output device number: 0
Type max. desired decimal places in table (0 to 8): 8

Type list of simple interest rates, expressed as percent,
ending list by typing digit 0:
 5., 5.25, 7.5, 10, 0

                    EFFECTIVE INTEREST

 SIMPLE   QUARTERLY   MONTHLY      360        365      CONTIN-
INTEREST    COMP.      COMP.     DAY YR.    DAY YR.     UOUS

 5.       5.0945337  5.1161897  5.1267447  5.1267496  5.12711
 5.25     5.3542668  5.3781886  5.3898527  5.3898583  5.39026
 7.5      7.7135866  7.7632599  7.787573   7.7875846  7.78842
10.      10.381289  10.471306  10.515557  10.515578  10.51709
READY
```

Run 2. Shows comparisons of rates entered.

```
                    INTEREST

Annual nominal interest (%)                5.00
Compounded how many times annually?        360
Number of times compounded?                360
Credited quarterly?                        NO
Amount on deposit?                     $   1000.73

                  By Binomial                    By
                   Expansion              ((1+(I/M))^N) -1
----------------------------------------------------------------
Effective rate  =      5.127     %             5.127 %

Interest earned =       $51.30                  $51.31
----------------------------------------------------------------


                    INTEREST

Annual nominal interest (%)                5.00
Compounded how many times annually?        360
Number of times compounded?                360
Credited quarterly?                        YES
Amount on deposit?                     $   1000.73

                  By Binomial                    By
                   Expansion              ((1+(I/M))^N) -1
----------------------------------------------------------------
Effective rate  =      5.127     %             5.127 %

Interest earned =       $12.59                  $12.59
Interest earned =       $12.75                  $12.75
Interest earned =       $12.91                  $12.91
Interest earned =       $13.07                  $13.07

   TOTAL        =       $51.32                  $51.32
----------------------------------------------------------------
```

Run 3. Quarterly interest crediting may introduce small errors.

Don't Give Me Any Static

How to guard against static-electricity damage to MOS circuits.

Conductive rubber floor mat drains static electricity to cement floor, or can be grounded with conductive plastic grounding strap.

Jess Kanarek
President, Wescorp
Mountain View CA

Microcomputers would not be possible without MOS microcircuits. These tiny bits of silicon perform functions previously performed by entire assemblies. Furthermore, their low cost has led to the proliferation of microcomputers.

Every technological advancement, however, has a price in terms of requiring changes in procedures, equipment and materials. The price for the advantages of microcircuits is that they are sensitive to a common phenomenon—static electricity.

Static electricity invades every home, office and work area—unseen and often undetected. Anyone handling a MOS device, or an assembly containing one, must learn a whole new set of rules.

Fortunately, any technological problem is usually followed by development of equipment, materials and techniques needed to make it most usable. MOS manufacturers first required techniques and equipment for protecting MOS devices while they were being produced and shipped. This technology was passed along to companies that installed the microcircuits in their products. If damage persisted, a quality control engineer often visited the user's plant to recommend any needed changes in the way the microcircuits were being handled.

However, the growing number of companies using MOS microcircuits for the first time and, more recently, the individuals using these products caused the static electricity problem to outgrow the propagation of knowledge of how to control it. Many manufacturers, as well as users, learned only through failed products that they had a static problem.

The growing need for anti-static protection inevitably led to the creation of independent companies specializing in anti-static materials and equipment. Competition among them has led to lower costs and even to development of procedures, equipment and materials the chip manufacturers never envisioned.

The Problem

To understand anti-static protection, it is necessary first to understand the nature of the problem. Static electricity can be generated by friction between many sources—carpets, clothing, motors and even metals. Static electricity can even occur when any two materials make or break contact. Some combinations of materials generate more static electricity than others. There is a transfer of electrons, with one material coming away with negative charges and the other with positive charges. Static electricity is dissipated, and a spark can even occur when this phenomenon (known as "tribo-electric" effect) takes place.

The most threatening source of static electricity to ICs, however, may be from the person who contacts them. An electrostatic charge of less than 100 volts can "blow out" the thin layer of glass that insulates the microcircuit gate from the substrate. A person's body can generate as much as 50,000 volts.

The blowout can be seen under a microscope and is, of course, quickly determinable in a functional test. However, the static-induced microcircuit failures that should be of most concern to the microcomputer user are not the catastrophic failures that may occur but the permanent shifts in conditions that cause failures due to degradation of performance. These are not easily detected and may cause unexplained malfunctions of the system in which the microcircuit is being used. Standard functional tests may not turn up the fault. Therefore, computer-user frustration can be added to the toll taken by static electricity.

Some Solutions

Anyone who handles a microcircuit or microcircuit assembly must be properly attired, and any surface that he or the microcircuit contacts must be properly equipped with anti-static materials and devices.

Furthermore, static electrici-

ty can be a problem when the microcomputer, or any other product containing microcircuits, is being used. Temporary malfunctioning can be added to the more costly perils already cited.

Safeguards against low-level static electricity are built into the circuits by the manufacturer, but these are not totally effective. However, the environments in which MOS ICs may be used—air conditioning, thick carpets and clothes of synthetic materials being the most ominous elements—are conducive to static electricity peak voltages that can overcome these safeguards. Anyone who builds his own system or carries out his own maintenance should raise the level of protection to that of the equipment manufacturer.

A minimum protective device is a conductive floor mat under the feet. These come with a conductive plastic grounding strap, with alligator clips or snaps on both ends for draining static electricity to any convenient ground before it can build up.

Basic to any anti-static control is proper clothing. Synthetic materials are excellent producers of static charges. In industry, a cotton smock has become the basic attire for anyone coming into contact with microcircuits.

Even properly attired, a person can generate high-voltage static-electricity charges. These must be drained off before you touch the microcircuit assembly. You must be grounded as near as possible to the point of contact with the circuit. A wrist strap, one end of which is clamped to a grounded surface, is the most desirable item for accomplishing this.

Some companies using MOS circuits require that shoes be wrapped in conductive plastic to create a ground between the shoes and the floor. Anyone who has touched a brass doorknob after walking across a nylon carpet on a dry day knows the amount of static electricity that can be generated by the friction between the feet and the floor.

Conductive work station combines wrist strap (held snugly to wrist by Velcro fastener), conductive felt workbench cover and grounding strap with clip.

In no case should a carpet be on the floor where microcircuits are handled. If the carpet cannot be removed, it should be covered with a grounded conductive material.

The tops of tables, workbenches or desks (executives have been known to blow out microcircuits by proudly showing them to visitors) must be covered with conductive plastics before the microcircuit assembly is placed on them. Conductive polyurethane foam is often selected because it also protects the circuit from mechanical shock or abrasion.

When properly set up, your anti-static work area is composed of an unbroken chain of conductive items from you to the nearest ground. The wrist strap is clamped to the conductive bench top, which is connected by a grounding strap to the floor covering, which is, in turn, linked to the ground.

Housekeeping is especially important where microcircuits are handled. Plastic notebook covers, candy wrappers, cigarette packages and other alien items can carry and generate static electricity, and should be removed.

Because of the triboelectric effect, a microcircuit can be a party to generating its own lethal charge of static electricity if it is placed in a parts tray of another material. For this reason a line of standard conductive plastic parts trays is available. They range from simple one-level trays with two or three compartments to three-level "lazy Susan" revolving trays having up to 24 different-size compartments.

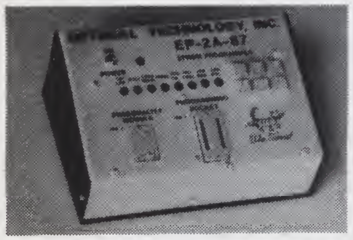
The air can also be a factor in controlling static electricity, which increases in inverse proportion to humidity. If possible, the relative humidity of a work area should be maintained at 40 percent or more. If the humidity is lower, greater handling precautions must be taken. The potential for static discharge is greatest if the plant interior environment is warm and dry, while the outside air is cold. Low-cost instruments also are available to detect and measure static electricity in an environment.

As a service to *Microcomputing* readers, Wescorp will sell basic anti-static devices by mail in single units. A conductive "work station," which combines a wrist strap, conductive-felt workbench cover and grounding strap, sells for $19.95. A conductive polyolefin floor mat measuring 24 × 32 inches sells for $35. Conductive polyolefin is more durable in such use and lies flat without curling at the edges. A wrist strap with Velcro enclosure sells for $5.60. A grounding strap is available for $6.

Check or money order should be sent to Wescorp, 1155 Terra Bella Ave., Mountain View CA 94043. Wescorp will pay shipping costs in the United States. ■

Assembler at Teledyne Microelectronics, Marina Del Rey CA, is properly attired for working on IC. She wears cotton smock, and wrist strap links her electrically to conductive plastic bench cover, which is grounded outside of photo. Note mouth mask, an unusual quality-control precaution.

Model EP-2A-87
EPROM Programmer



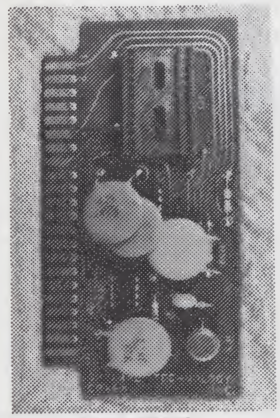
The Model EP-2A-87 EPROM Programmer has an RS-232 compatible interface and includes a 2K or 4K buffer. During the ON-LINE mode, another computer can down-load to the buffer. Only two easy-to-implement commands are available to an external computer. (Load buffer and read buffer.)

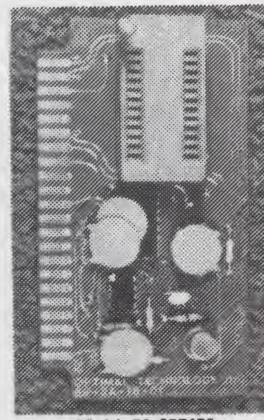
In the OFF-LINE mode, the EP-2A-87 will program, verify, test buffer, and load the buffer from the EPROM socket. During the programming cycle, the EPROM is checked before programming to insure that it is erased and after programming it automatically verifies that programming is correct. Power requirements are 115 VAC 50/60 Hertz at 15 watts.

Part No.	Description	Price
EP-2A-87-1	Programmer with 2K buffer	$525.00
EP-2A-87-2	Programmer with 4K buffer	600.00
	Non standard voltage option (220 v, 240 v, 100 v)	15.00
PM-0	Personality Module, programs TMS 2708	26.00
PM-1	Personality module, programs 2708	26.00
PM-2	Personality module, programs 2732	31.00
PM-3	Personality module, programs TMS 2716	26.00
PM-4	Personality module, programs TMS 2532	31.00
PM-5	Personality module, programs 2716, TMS 2516	16.00
PM-6	Personality module, programs 2704	26.00
PM-7	Personality module, programs 2758, TMS 2508	16.00
PM-8	Personality module, programs Motorola MCM68764	34.00
MS-XX	Disk driver software	27.50

Optimal Technology, Inc.
Blue Wood 127
Earlysville, Virginia 22936
Phone (804) 973-5482

EPROM PROGRAMMERS





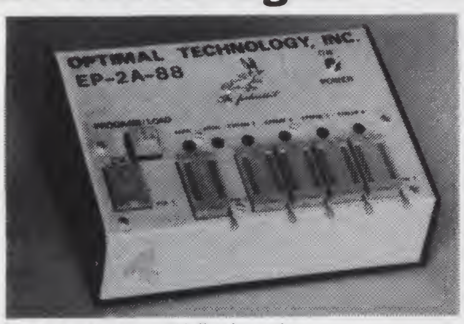
EP-2A SERIES
- PROGRAMS 2708 and 2716 EPROMS
- Price $59.95 Assembled and Tested
- Kit price $49.95
- Includes Connector

EP-2A-78 SERIES
- PROGRAMS 2708, 2716, 2758, TMS 2716 and TMS 2532 EPROMS
- TEXTOOL ZERO FORCE SOCKET
- Price $79.95 Assembled and Tested
- Includes Connector

Software available for the Rockwell AIM-65, MOS Technology KIM-1, Synertek SYM-1, Motorola D2, RCA VIP and many other single board computers that use the 6502, 6800, 8080/85, Z-80, 1802, F-8 and 2650 CPU's. Stock. Specify one set of software.

Optimal Technology Inc.
Blue Wood 127
Earlysville, VA 22936 U.S.A.
Phone (804) 973-5482

Model EP-2A-88
EPROM Programmer



Fast as Jackrabbits . . . Well, almost!

In Australia, two rabbits can reproduce over 13 million offspring in 3 years . . . at 105 seconds per 2716, the **EP-2A-88** can reproduce 1,892,160 EPROMS in 3 years. **Single push button control, the EP-2A-88 checks if EPROMS are erased, programs and verifies. It also checks for defective EPROMS.**

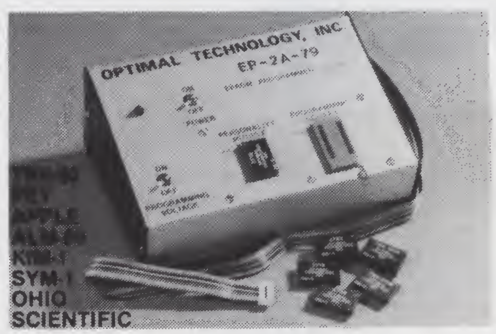
Two basic models are available, The EP-2A-88-1 will accept Copy (CM) modules for the 2758, and 2716 EPROMS. The EP-2A-88-2 will accept copy modules for the 2716, 2732 and TMS 2532 EPROMS. Power requirements are 115 VAC 50/60 Hertz at 15 watts.

Part No.	Description	Price
EP-2A-88-1	EPROM Programmer	$450.00
EP-2A-88-2	EPROM Programmer	450.00
CM-50	Copy Module for 2716, TMS 2516 EPROMS	25.00
CM-70	Copy Module for 2758 EPROMS	25.00
CM-20	Copy Module for 2732 EPROMS	25.00
CM-40	Copy Module for TMS-2532 EPROMS	25.00
	Non-Standard Voltage Option (220 v, 240 v, 100 v)	15.00

Optimal Technology, Inc.
Blue Wood 127, Earlysville, Virginia 22936
Phone (804) 973-5482

✓29

Model EP-2A-79
EPROM Programmer



Software available for F-8, 6800, 8085, 8080, Z-80, 6502, 1802, 2650,6809, 8086 based systems.

EPROM type is selected by a personality module which plugs into the front of the programmer. Power requirements are 115 VAC 50/60 Hz. at 15 watts. It is supplied with a 36-inch ribbon cable for connecting to microcomputer. Requires 1½ I/O ports. Priced at $155 with one set of software. (Additional software on disk and cassette for various systems.) Personality modules are shown below.

Part No.	Programs	Price
PM-0	TMS 2708	$15.00
PM-1	2704, 2708	15.00
PM-2	2732	30.00
PM-3	TMS 2716	15.00
PM-4	TMS 2532	30.00
PM-5	TMS 2516, 2716, 2758	15.00
PM-8	MCM68764	33.00

Optimal Technology, Inc.
Blue Wood 127, Earlysville, Virginia 22936
Phone (804) 973-5482

Is Your Computer Holding You Back?

If so...it's time you expand your capabilities with a Vista V-200 double density minifloppy disk system.

Features:

- Storage capacity ranges from 200K bytes to 1.2 megabytes.
- All Lifeboat software is compatible with the Vista V-200.
- The Vista Exidy 200 does not require a S100 expansion box.
- The Vista V-200 is totally compatible with:
  Processor Technology's SOL-20
  North Star's Horizon
  Exidy's Sorcerer
  Ismai's 8080 systems.
- Immediate Delivery.
- 120 Day Warranty.
- The complete system is available for as low as **$695.**

Each V200 Minifloppy Disk System Includes:

■ Minifloppy disk drive(s)—includes dc power regulator board, case and internal power supply. ■ S100 bus controller card—plugs into your computer and controls up to 3 double-density double-headed disk drives. ■ I/O cable—connects controller to drive(s). ■ System software—Vista CP/M (VOS) Disk Operating System and BASIC-E compiler (CBASIC optional) recorded on 5¼'' diskettes. ■ Operating/instruction manuals—complete hardware software documentation describing your V200 system. ■ Each 5¼'' diskette holds up to 400K bytes.

Complete Systems Approach

If you have a Z-80/8080-based microcomputer with an S100 Bus, our V200 Minifloppy Disk System is everything you need to expand you computer. Completely software/hardware compatible, the V-200 is a double density system capable of running up to 3 double-density double-headed disk drives.

The fast-access, on line storage of the V200 system gives you instantaneous program loading and dumping; sequential and random file access; context editing of programs and text; dynamic debugging of programs; program assembly; batch processing and much more. With your console, computer and about 24K bytes of main memory, the V-200 gives you a complete disk-oriented computer system with automatic bootstrap loading. Just turn on the power, reset, and you're ready to run.

For Further Information Call our Toll Free Number **800-854-8017**

Vista

✓146

The Vista Computer Company 1401 Borchard Street • Santa Ana, California 92705 • 714/953-0523

Take a Letter, Ms. TRS-80

Not exactly a word processor, this "secretary" program will compose letters for you.

Dr. Jack N. Adams
209 South Lincoln
Jerome ID 83338

Many of you would like to use your computer to write letters and documents. There are many programs, such as Electric Pencil, that you could use, but they cost money that could be used for better things, such as a printer. So here is a free and simple letter writer. You could call it a word processor, but that would be overstating its ability.

This program, written in TRS-80 Disk BASIC, will allow you to write a letter, right-justify, file on disk, add lines at end of letter, modify any lines by retyping and print a hard copy. After the flak I took on my last article ("Let's Go Flying," *Microcomputing*, April 1979, p. 68), I will tell you how to rewrite the program in the language of your choice. It is not possible to show a program run because of the type of data entry (INKEY$) used. Therefore, I will include a "manual" for use of the program.

This program was written as part of a mail-list program. The object of the program is to write a letter, file it on a disk and then call the letter up and send it to a select group on the mail-list. By the way, I've used consecutive line numbers for your convenience. Remark statements have not been referenced by line number and may be dropped without affecting the program.

Line by Line

In line 40 the CLEAR (4500) is used to clear enough room for 55 lines of 80 characters. The program will allow 56 lines of 128 characters. If you are going to write long letters with long line lengths, you will need to change this to a higher value. The DEFINT H–O,P,S,Y,Z,E is used to define as integer all variables beginning with the letters listed. Some of this is a carry-over from the other programs and may not be necessary in this program.

Line 50 reads into the array C$(I) the values in line 60. These letters in the DATA statement are the letters from the "Menu." This allows input of only the choices available. Now comes the real trouble.

I've used the INKEY$ function extensively. The first place you see it is in line 130, which says "GOSUB 1280." If you do not have a TRS-80, you may want to completely forget about all of the lines that I have used for input. Some BASICs have the INP function that you may be able to use in the place of the INKEY$. I'll take you through to explain what is going on.

Line 1280 first sends you to line 1300, which locates the cursor and sets the value in P for the PRINT @ statements. If you do not have a TRS-80 but do have cursor control, you can set the value of P by some calculation. Next, the GOSUB to 980 sets A$ to a null. Line 990 prints character string 143 at screen position P. This is a graphic character. You could use 45 (minus) or 95 (underline) just as well. L is then set to 1.

The next line, 1000, looks at the keyboard to see if a key is pressed, and, if it is, it sets A$ to the keyed value. It checks to see if it is a null, and, if it is not, it prints the value of A$ at P. This allows a single key input. If an input occurred, you would be returned. However, if a key was not pressed, the program would go to line 1010.

In this line, L is incremented by 1 and then checked to see if L is 16. If L is less than 16, the program will go back to the keyboard in line 1000 to look for another input. If L is 16, line 1020 will print a space at P and go into a FOR loop from 1 to 10, and then back to line 980. This causes a blinking cursor.

The variable "I" in line 130 is set to 1. In line 140 a checking loop starts using "I" as the counter. First C$(I), which is the letter "C," is checked to see if it is the same as A$. If it is not, "I" is increased by 1 and the second C$(I) is checked.

This goes on for eleven tries.

If no match occurs, the program goes back to line 130 and then to subroutine 980 for another input. This prevents any input except for the eleven letters in the DATA statement in line 60. If a match occurs, you will be sent to line 150 for the ON-GOSUB statement to send you to the chosen subroutine.

The other subroutine for input is first encountered in lines 190 and 200. (By the way, the CLS is Radio Shack for clear screen.) In line 190 the variables "O" and "I" are set to key the subroutine. "O" is the length of the input allowed, and "I" is the subscript for the variable B$(I). The subscript for B$ is a holdover from the other programs and could be dropped.

In line 200 you're sent to the subroutine for multiple entry in line 1040. The cursor is turned on by PRINT CHR$(14); then B$(I) is set to a null. Line 1050 looks for an input for A$ from the keyboard. If none occurs, it will look again. When one occurs, it goes to line 1060. In this line, if FL = 1 (the input routine for creating a letter sets this to this value), then a check is made to see if the entry is ASC(44) or a comma.

It is necessary to check for a comma in this subroutine because I used serial files to file the letter. If you have a comma in the text, the program will mistake this for the end of a variable on input. As a result, the text read from a disk will be messed up. If you use random files, this can be changed.

The next line (1080) is used to check for a back space (CHR$(8)). If A$ is a back space, the string will be shortened by one character. Line 1090 checks for CHR$(24). This allows you to erase the entire input of the line by pressing shift-back space.

The next line checks to see if the entered value of A$ is between ASC(32) and ASC(90). This limits the entry to the uppercase alpha as well as the numbers and symbols. If you have modified your TRS-80 to use lowercase, you will need to increase the upper limit to 122 to include the small "z." You may even want to go to 126. Line 1130 limits the length of B$(I) to the length specified in the variable "O." When the entry is this length, the only input accepted is the back space or "ENTER."

In line 1120, the value of A$ is printed on the screen and A$ is added on the end of B$(I). Then the program goes back to line 1070 for another letter until a carriage return (CHR$(13)) turns off the cursor (CHR$(15)) and returns you to the subroutine that sent you to line 1040.

Meanwhile, back at line 200, the program adds "/TXT" to the name entered, prints the name and asks if this is correct. This extension is necessary to avoid the possibility of messing up programs or files other than those created by this program.

Line 210 sends you back to line 1280 for a single key entry. This is checked—if "Y," you will proceed; if "N," then you must go back for another try at a name; if "E," forget it, or else try another entry. Lines 220 to 250 may need to be rewritten for your BASIC. Here you set the disk number, check the entry for limits of possibility and then kill the file. C-BASIC running under CP/M would be much different:

```
220 PRINT
    INPUT "Which Drive is it on (A-D)";A$
    IF A$<"A" OR A$> "D" THEN 220
    NA$ = A$ + ":" + NA$
    OPEN NA$ AS 1
    DELETE 1
    CLOSE 1 : RETURN
```

To be honest with you, I haven't tried these lines of code, so please don't be too angry with me if I goofed.

The next line, 270, is just the end. You could simply have an END statement here. (However, in C-BASIC you would want to use STOP.) Lines 290 to 340 are used to set the control for line length (number of characters per line) and length of page. The limits for line length are 1 to 128. The length of a page is 6 to 66. You may want to change these values.

In both cases a carriage return (ENTER) will retain the old values. PP is the variable name for page length and "OO" is the variable for line length. The tab position is displayed in this subroutine (TA) but not set here.

Line 360 is called several places in the program. The function of the line is to calculate the number of spaces needed at the top of the page. This allows the program to control the placement of the letter vertically on the page. The variables used are: TP for top of page, PP for page length and NN for number of lines. The NN variable is set when the letter is created.

The lines 380 to 420 are used to input the letter. Starting with the first line, "FL" is set to 1 to be a flag for the input subroutine to eliminate the comma. Then you are routed to 290 to set the variables for line and page length. Next come a clear screen (CLS) and printing of the header. The variable O is set to the line length (OO). The next line starts a loop for entry of the letter. II is incremented by one, the line number is printed for reference, a line of minus signs is printed and then the program goes to subroutine 1300 to locate the cursor, which is then back-spaced to the start of the minus-sign line. Then the program goes to 1040 for the input

```
5/22/79

WAYNE GREEN
MICROCOMPUTING - KILOBAUD
PETERBOROUGH  NH  03458

DEAR WAYNE:

THIS IS AN EXAMPLE OF A LETTER TYPED ON MY TRS-80 TO TEST THE "WRITE"
PROGRAM.  THE PARAMETERS USED WERE:  LINE LENGTH - 75  AND PAGE LENGTH 66.
I WILL PRINT IT IN THE ORIGINAL FORM AND ALSO IN THE RIGHT-JUSTIFIED
FORM.  THE USER HAS THE OPTION TO SET THE TABULATION AFTER

THE TEXT CAN BE ALTERED AFTER THE ORIGINAL WRITING AND CAN BE RE-FILED
AS A NEW FILE OR CAN REPLACE THE ORGINAL FILE.  PAGE LENGTH CAN BE RE-
DEFINED AFTER THE LETTER IS WRITTEN.  THE PROGRAM AUTOMATICALLY PRINTS THE
TEXT IN A VERTICALLY-CENTERED FORMAT.

SINCERELY,


DR. JACK N. ADAMS
209 SOUTH LINCOLN
JEROME   ID  83338
```

```
5/22/79

WAYNE GREEN
MICROCOMPUTING - KILOBAUD
PETERBOROUGH  NH  03458

DEAR WAYNE:

THIS IS AN EXAMPLE OF A LETTER TYPED ON  MY  TRS-80  TO  TEST  THE  "WRITE"
PROGRAM.  THE PARAMETERS USED WERE:  LINE LENGTH - 75  AND PAGE LENGTH  66.
I WILL PRINT IT IN THE  ORIGINAL  FORM  AND  ALSO  IN  THE  RIGHT-JUSTIFIED
FORM.  THE USER HAS THE OPTION  TO  SET  THE  TABULATION  WHEN  HE  PRINTS.

THE TEXT CAN BE ALTERED AFTER THE ORIGINAL  WRITING  AND  CAN  BE  RE-FILED
AS A NEW FILE OR CAN REPLACE THE ORGINAL FILE.   PAGE  LENGTH  CAN  BE  RE-
DEFINED AFTER THE LETTER IS WRITTEN.  THE PROGRAM AUTOMATICALLY PRINTS  THE
TEXT IN A VERTICALLY-CENTERED FORMAT.

SINCERELY,


DR.  JACK N. ADAMS
209 SOUTH LINCOLN
JEROME   ID  83338
```

Letter-writing printout. a) Before correction and justification. b) After correction and justification.

Write letter program manual.

CUSTOM WRITING PROGRAM (W-Option)

This program will allow you to write a letter, right justify, file on disk, call up from disk, and print a trial copy. The display will give the following choices:

<C>REATE NEW LETTER
<J>USTIFY (RIGHT MARGIN)
<S>ET PRINTER CONTROLS
<P>RINT HARD COPY
<R>EAD LETTER FROM DISK
<F>ILE LETTER ON DISK
<A>LTER TEXT
<K>ILL LETTER FILE
<E>XIT PROGRAM
<I>NCREASE LENGTH OF LETTER
<T>YPE OUT ON SCREEN

You enter the letter for the program you wish to use. The program will return to this display until you enter "E" to EXIT the program and return to the main "MENU".

CREATE NEW LETTER (C-Option)

This program allows you to enter a letter. The program leads you step by step. First, you are asked:

HOW MANY LINES PER PAGE?-

This sets the length of the page for your letter. Eleven inch paper will have 66 lines. If you want to print two or more letters on a page you can divide the number of letters per page into 66 to get the value. Example: If you want to print two letters per page you would enter 33. CAUTION: You should remember that the address will take up 5 lines on the printed letter. The number of lines should allow for a few lines between letters.

The program will then ask:

HOW MANY CHARACTERS PER LINE?-

This is where you enter the length of the line. Normally you would enter 60 to 80 at this point. The limits are 1 to 128. However, a line length of 128 would result in very small print. The next step is the entry of the letter. The screen will display:

ENTER LETTER:
1. --

The line of "-" is an indicator of the line length. If the line length is over 59 characters, the line will "wrap around" to the next line. The number (1 in this case) is a line number for reference and does not appear in the printed copy. It serves as a reference for later use in altering (A-Option) the letter.

You can proceed to type the letter. When you get to the end of the line the keyboard of the computer will lock preventing futher entry. At this point you must back space (using the backspace arrow) until you get to the last complete word, then press ENTER. The program will respond with:

2. --

You then enter each line as you wish until you have completed your letter. To exit this mode, you enter END at the start of the line following the last line of the letter. The program will take you back to the main display.

JUSTIFY RIGHT MARGIN (J-Option)

This portion of the program allows the user to justify the right margin of a letter that has been created with the C-Option. First, you will be asked:

DO YOU WANT TO RIGHT JUSTIFY THIS LETTER?(Y/N)

If you respond with "N" the program returns to the main display. The program checks at this point to see if this letter has been called up from a disk file. If it has, it will go to the SET PRINTER CONTROLS (S-Option) to reset page length and line length. The screen will then display the first line of the letter and ask:

1. THIS IS A TEST OF THE WRITE PROGRAM TO SEE IF IT WILL

DO YOU WISH TO RIGHT JUSTIFY THIS LINE? (Y/N)

If you respond with "N", the next line will be displayed. If you answer "Y" the program will justify the line and then display the justified line and say:

THE LINE NOW READS:
1. THIS IS A TEST OF THE WRITE PROGRAM TO SEE IF IT WILL
TO CONTINUE PRESS >ENTER<

The program will display each line until all lines of the letter have been either justified or not. The program then returns to the main display.

TYPE OUT ON SCREEN (T-Option)

This option is used to type out on the computer screen the letter written with the "C" option. Each line is displayed with the reference number. You can stop the display by pressing the @ key while holding the shift key down. The display will be continued by pressing any key. At the completion of the display of the letter the screen will say:

PRESS ENTER TO CONTINUE:

When you press ENTER the program will return to the main display.

of the line.

After the input you are routed to subroutine 360 to adjust the top of page (TP). Then in line 400 a check is made to see if you have entered 56 lines. If you have, FL flag is set to 0 and you are returned to the main display. If not, the next line checks to see if "END" has been entered. If it has, II is reduced by one, FL is reset, you go to line 360 again, and then return. Line 420 says, "Do it again."

Potential Problem Areas

Line 610 could present a problem. This line contains a check of the line to be justified. If the line is short enough to need more than 84 percent justification, the program will refuse. I did this to avoid the possibility of an endless loop. Line 620 is a check to see if any justification is needed.

The actual process takes place in line 650. Going from right to left through the string BB$(II), a check is made for a space (CHR$(32)). If one is found, another space is added, so do not leave a space hanging on the end of the line or you will have two of them! Then the count of spaces needed (JU) is reduced by one. The compensator for spaces (YY) is increased by one and a check to see if you are finished is made. Line 670 checks to see if you have finished on the first trip through the string. If not, do it again.

The filing routine starting in line 740 is one place that could be a problem with other languages. I have used serial files because they are simple. In this case, I think they may use less space. Line 820 opens the file, 830 writes the first four variables on the file and then 840 writes the text of the letter on the file. C-BASIC might write the file as:

```
    IF END #1THEN 855
820 OPEN NA$ AS 1 RECL 128
830 PRINT # 1,1;PP,NN,TP,OO
    FOR X = 2 TO NN + 1
    PRINT # 1,X;BB$(X)
    NEXT X
850 CLOSE #1 : RETURN
855 CREATE NA$ AS 1 RECL 128
    GOTO 830
```

The read routine starting in line 870 would be the same as the filing one except READ would be used in place of PRINT and the CREATE would not be necessary. The END trap could be used to avoid jumping out of the program if the file did not exist.

The printing subroutine starting in line 1180 has several things in it that could give you trouble. Line 1220 limits the tab setting to a maximum of 64. This is all that you can tab a Radio Shack printer. Line 1240 POKEs the value for the page length into the memory location of the TRS-80. In another BASIC you probably would not need to do this. Then a PEEK is made to see if the printer is on. Again, this is not needed unless you are using the TRS-80.

Line 1250 prints carriage returns for the spaces at the top of the letter. You could use the following (with C-BASIC):

```
1250 FOR X = 1 TO TP
     PRINT
     NEXT X
```

The next line (1260) calculates the number of lines at the end of the letter (OP). Line 1270 prints out the text of the letter and then, using LPRINT STRING$ (OP,138), prints the blank lines at the end of the page. You could use a FOR loop as shown above to do this.

Credit should be given to Radio Shack for the two subroutines for INKEY$. These two routines are very useful for inputting without having to press "ENTER" all the time. One pointer on listing programs written in TRS-80: If you place a carriage return (down arrow) in the line, the printer will do just that on output. If these carriage returns are placed between statements on the line, they will have no effect on the program run. ■

SET PRINTER CONTROLS (S=Option)

This program displays the control settings and gives asks you to change them. The program asks:

```
PRINT CONTROLS ARE:

LINES PER PAGE........66
CHARACTERS PER LINE...72
TAB POSITION..........10

HOW MANY LINES PER PAGE?
```

You will then enter a number between 6 and 66 followed by pressing ENTER. The program will reject any number out of this range by printing a question mark (?) and wait for your entry. If you press ENTER without entering a value, the old value is retained. The program will then ask:

```
HOW MANY CHARACTERS PER LINE?
```

You will enter the number of characters you wish to have the line length to be. The program will reject any entry other than a number between 1 and 128. Again, if you press ENTER without inputing a value, the old value will be retained. The program will then return to the main display.

PRINT HARD COPY (P-Option)

This program will print on the Radio Shack Line Printer a copy of a letter created with the C-Option. If you are calling up a letter that has been saved, the program will presume that the printer controls used when you created the letter are to be used. If you do not wish for this to be the case, you should set these controls by using the S-Option before printing. The program will ask:

```
THE PRESENT TAB SETTING IS 10, IS THIS OK?(Y/N)
```

If you respond with "N", the program will ask:

```
WHAT IS THE TAB POSITION?
```

You may enter any between 0 and 64. The program will reject any other entry. The program will then say:

```
PRESS ENTER TO CONTINUE, E TO EXIT
```

If you press "E" the program will return to the main display. If you press ENTER the program checks to see if the printer is turned on. If it is not it will display:

```
** PRINTER NOT READY **

PRESS ENTER TO CONTINUE, E TO EXIT
```

Printing of the letter will proceed after you have turned on the printer and pressed ENTER. The program will center the letter vertically. You will need to use the print size adjustment and tab to locate the letter left to right on the paper. After printing the letter the program will return to the main display.

READ LETTER FROM DISK (R-Option)

This program will read a letter from the disk that has been filled there (using the F-Option) after you created it using the C-Option). The program will ask:

```
READ LETTER FROM DISK
ENTER NAME OF LETTER
```

You will then enter a name for this letter. This may be up to eight characters or numbers but must start with an alpha character. This name must be unique or the program will write this letter on the disk in place of the other one. The program will then ask:

```
ON WHICH DRIVE IS THE FILE (0-3)?
```

When you press a number between 0 and 3 the program will print the file name on the screen:

```
THE FILE NAME IS TEST/TXT:1
** READING DISK **
```

The program adds the "/TXT:1" to indicate that it is a text file residing on drive 1. This is to prevent overwriting one of the programs. If the file does not exist on the drive you chose, you will get an error message to this effect. You need only type RUN and press ENTER to restart the program. When the letter has been read into the computer from the disk, it will be TYPED onto the screen. This aids in identification to be sure it is the correct one. You should keep a copy of these letters along with their file names. It is possible to stop the printing to look at the letter by pressing the @ key while holding the shift key down, as in the type (T-Option) program.

FILE LETTER ON DISK (F-Option)

This program will file a letter on the disk that has been created or modified in the program. The program will ask:

```
WHAT DO YOU WANT TO CALL THIS LETTER?
```

You may enter a unique name up to eight characters long, the first of which must be alpha. The program will then print the name to which it has added "/TXT" and ask:

```
THE FILE WILL BE TEST/TXT
IS THIS ALRIGHT?
```

If you answer "N", the program will ask for a new name. If you answer "Y", the program will ask:

```
ON WHICH DRIVE DO YOU WANT TO FILE (0-3)?
```

After you have indicated on which drive you want the file, the program will say:

Program listing.

```
10 REM - LETTER WRITING PROGRAM -
20 REM - MAY 1979 -
30 REM - DR. JACK ADAMS : JEROME, ID 83338
40 CLEAR(4500):DEFINTH-O,P,S,Y,Z,E:DIMBB$(56):DIMC$(11)
50 FORI=1TO11:READC$(I):NEXTI
60 DATAC,J,S,P,R,F,A,K,E,I,T
70 TA=10
80 CLS:PRINT@18,"CUSTOM WRITING PROGRAM"
90 PRINT@128,"<C>REATE NEW LETTER":PRINT"<J>USTIFY (RIGHT MARGIN)":PRINT"<S>ET PRINTER CONTROLS
   PRINT"<P>RINT HARD COPY"
100 PRINT"<R>EAD LETTER FROM DISK":PRINT"<F>ILE LETTER ON DISK":    PRINT"<A>LTER TEXT"
    PRINT"<K>ILL LETTER FILE"
110 PRINT"<E>XIT PROGRAM":PRINT"<I>NCREASE LENGTH OF LETTER":PRINT"<T>YPE OUT ON SCREEN"
120 PRINT"":PRINT"SELECT YOUR OPTION";
130 GOSUB1280:I=1
140 IFC$(I)=A$THEN150:ELSEI=I+1:IFI<12THEN140ELSE130
150 ONIGOSUB380,550,290,1180,870,740,440,180,270,1310,1140
160 GOTO80
170 REM - KILL SUBROUTINE -
180 CLS:PRINT@18,"KILL LETTER FILE":PRINT""
190 PRINT"WHAT IS THE NAME OF THE FILE?";:O=8:I=0
200 GOSUB1040:NA$=B$(I)+"/TXT":PRINT"":PRINT"THE FILE NAME IS ",NA$:PRINT""
    PRINT"IS THIS THE CORRECT FILE (Y/N)?";
210 GOSUB1280:IFA$="Y"THEN250:ELSEIFA$="N"THEN190:ELSEIFA$="E"THENRETURN:ELSE210
220 PRINT"":PRINT"WHICH DRIVE IS IT ON (0 - 3)?";:GOSUB1280
230 IFVAL(A$)<0ORVALA$>3THEN220ELSENA$=NA$+":"+A$
240 PRINT:PRINT"TO KILL THIS FILE PRESS ENTER (E TO EXIT)";:GOSUB1280:
    IFA$="E"THENRETURNELSECLS:PRINT@276,"** KILLING FILE **"
250 KILLNA$:RETURN
260 REM - EXIT -
270 CLS:PRINT@276,"** RETURNING TO TRSDOS **":CMD"S"
280 REM - SET SUBROUTINE -
290 CLS:PRINT"CONTROLS NOW ARE:":PRINT"":PRINT"LINES PER PAGE.......";PP
    PRINT"CHARACTERS PER LINE...";OO:PRINT"TAB POSITION.........",TA:PRINT""
    PRINT"HOW MANY LINES PER PAGE?";:O=2:I=0
300 GOSUB1040:IFB$(0)=""THEN320ELSEIFVAL(B$(0))>66ORVAL(B$(0))<6THENPRINT"?":GOTO300
310 PP=VAL(B$(0)):GOSUB360
320 PRINT"":PRINT"HOW MANY CHARACTERS PER LINE?";:O=3
330 GOSUB1040:IFB$(0)=""THEN340ELSEIFVAL(B$(0))>128ORVAL(B$(0))<1THENPRINT"?"
    GOTO330:ELSEOO=VAL(B$(0))
340 RETURN
350 REM - TOP OF PAGE -
360 TP=INT((PP-NN)/2):RETURN
370 REM - CREATE SUBROUTINE -
380 FL=1:II=0:GOSUB290:CLS:PRINT"ENTER LETTER ":O=OO:I=0
390 II=II+1:PRINTUSING"##. ";II;:FORR=1TOOO:PRINT"-";:NEXTR:GOSUB1300:P=P-OO-1:
    PRINT@P,"";:GOSUB1040:BB$(II)=B$(I):NN=II:GOSUB360
400 IFII>55THENFL=0:RETURN
410 IFBB$(II)="END"THENII=II-1:NN=II:FL=0:GOSUB360:RETURN
420 GOTO390
430 REM - ALTER SUBROUTINE -
440 IFNN<1THENRETURN
450 CLS:PRINT"WHICH LINE DO YOU WANT TO CHANGE? (1 TO ",NN,")";:O=2:I=0:GOSUB1040
460 IFB$(I)="E"THEN RETURN
470 NU=VAL(B$(I)):IFNU<1ORNU>NNTHEN450
480 PRINT"LINE ";NU;" READS:":PRINTBB$(NU):PRINT:PRINT"IS THIS THE CORRECT LINE? (Y/N)";:
    GOSUB1280:IFA$="N"THEN450
490 PRINT"":PRINT"DO YOU WANT TO RETYPE THIS LINE? (Y/N)";:GOSUB1280:IFA$="Y"THEN 500:
    ELSEIFA$="N"THEN450:ELSE480
500 PRINT:PRINTBB$(NU):PRINT"ENTER NEW LINE ";NU:O=OO:I=0:GOSUB1040:BB$(NU)=B$(0)
510 PRINT"":PRINT"LINE ";NU;" NOW READS:":PRINTBB$(NU):PRINT"":PRINT"IS THIS CORRECT (Y/N)?";
520 GOSUB1280:PRINT"":IFA$="Y"THEN450
530 GOTO500
540 REM - JUSTIFY SUBROUTINE -
550 IFNN<1THENRETURN
560 CLS:PRINT"DO YOU WANT TO RIGHT JUSTIFY THIS LETTER? (Y/N)";:GOSUB1280:PRINT""
570 IFA$="N"THENRETURN
580 IFOO<1THENGOSUB290:CLS
590 FORII=1TONN:PRINTUSING"###. ";II;:PRINT" ";BB$(II):PRINT"     ";:FORI=1TOOO:PRINT"-";:NEXTI:PRINT
600 PRINT"DO YOU WISH TO RIGHT JUSTIFY THIS LINE? (Y/N)";:GOSUB1280:PRINT"":IFA$="N"THEN710
610 JU=OO-LEN(BB$(II)):YY=0:IFJU>INT(LEN(BB$(II))*.16)THENPRINT"":PRINT"**  SHORT LINE  **":GOTO710
620 IFJU<1THEN680
630 LN=LEN(BB$(II))
640 FORX=LNTO1STEP-1
```

TO WRITE THE FILE PRESS ENTER (TO EXIT PRESS E)

At this point you may exchange the disk on the drive you specify so that you could keep all the letters on a special disk. It is possible to write the disk file on any disk, but it may be better to have separate disk for this job. If you press "E" the program will return to the main display without filing the letter on disk. If you press ENTER the program will display:

** WRITING FILE **

When the letter has been written on the disk, the program will return to the main display.

ALTER TEXT (A-Option)

This program allows you to retype any line of the letter. This allows correction of misspelled words, correct line length or completely changing a line. It is possible to add lines at the end of the text by using the Increase Length of Letter (I-Option). It should be mentioned that a blank line will occur if you press ENTER when asked to enter the line.
The program will ask:

WHICH LINE DO YOU WISH TO ALTER (1 TO 32)

You would indicate the line number you think is correct. If you wish to exit the program you could press "E" and then ENTER to return to the main display. The program will display the line and ask:

LINE 21 NOW READS:
THIS IS A TEST OF THE WRITE PROGRAM TO SEE IF IT WILL
IS THIS THE CORRECT LINE? (Y/N)

If you answer "N", the program will ask for the line you wish to alter. If you answer "Y", the program will ask:

DO YOU WANT TO RETYPE THIS LINE? (Y/N)

A "N" at this point will return you to asking to enter the line you wish to alter. If you press "Y" the program will say:

ENTER NEW LINE 21:

You will then retype the complete line. After you type the line and press ENTER, the program re-displays the line and asks if it is correct:

LINE 21 NOW READS:
THIS IS A TEST OF THE WRITE PROGRAM TO SEE THAT IT WILL
IS THIS CORRECT (Y/N)?

If you respond with "N", the program will give you a chance to retype the line again. A "Y" will return you to the starting question in A-Option. You may exit this program by entering "E" and pressing ENTER when the program asks which line you want to change. You would return to the main display.

KILL LETTER FILE (K-Option)

This program allows you the option to delete a file from a disk. It is only possible to kill a "/TXT" file using this method. Great care should be used when you use this program! The screen will say:

KILL LETTER FILE

WHAT IS THE NAME OF THE FILE?

You would then type in the exact name of the file. The program will add the "/TXT" to the file. This prevents killing any file other than a letter file created with this program. Next, the program will ask:

THE NAME OF THE FILE IS TEST/TXT
IS THIS THE CORRECT FILE (Y/N)?

If you press "E", the program will return to the main display. If you press "N", the program will ask for the file again. The program then asks:

WHICH DRIVE IS IT ON (0-3)?

You then enter the correct drive. The program will display then display the message:

TO KILL THIS FILL PRESS ENTER (E TO EXIT)

If the file is not on the drive you enter, an error occurs. To restart the program type RUN and press ENTER. The program will kill the file if all goes well above and then return to the main display.

EXIT PROGRAM (E-Option)

When you press "E" while in the main display the program will clear the screen and print:

** RETURNING TO TRSDOS **

This returns you to the Disk Operating system.

INCREASE LENGTH OF LETTER (I-Option)

This program allows you to add lines of text to the end of the letter. The program looks up the current letter's length and starts by displaying:

ENTER ADDITIONAL LINES:

33. ---

You can then enter the lines you wish to add in the same manner as in the Create (C-Option) program. When you have completed all the lines you intend to add, typing END as in the Create (C-Option) will terminate this program. If there is no text in the program from a letter previously entered, the program will reject entry to this program.

```
650 IFMID$(BB$(II),X,1)=CHR$(32)THEN BB$(II)=LEFT$(BB$(II),X)+" "+RIGHT$(BB$(II),((LN-X)+YY)):
    JU=JU-1:YY=YY+1:IFJU<1THEN680
660 NEXTX
670 IF JU<>0THEN630
680 PRINT"THE LINE NOW READS:":PRINTUSING"###. ";II;:PRINT" ";BB$(II)
690 PRINT"TO CONTINUE PRESS >ENTER< ";:GOSUB1280
700 PRINT""
710 NEXTII
720 RETURN
730 REM - FILE SUBROUTINE -
740 IFNN<1THENRETURNELSE:CLS:PRINT"WHAT DO YOU WANT TO CALL THIS LETTER?";
    :O=8:I=0:GOSUB1040
750 NA$=B$(I)+"/TXT":PRINT:PRINT"THE FILE NAME WILL BE ";NA$
760 PRINT"":PRINT"IS THIS ALRIGHT? ";:
770 GOSUB1280:IFA$="N"THEN740ELSEIFA$="E"THENRETURN
780 PRINT"":PRINT"ON WHICH DRIVE DO YOU WANT TO FILE (0-3)?";:
790 GOSUB1280:NA$=NA$+":"+A$
800 PRINT"":PRINT"TO WRITE THE FILE PRESS ENTER (TO EXIT PRESS E)";:
810 GOSUB1280:PRINT"":IFA$="E"THENRETURN
820 CLS:PRINT@276,"** WRITING DISK **":OPEN"O",1,NA$
830 PRINT#1,PP;",";NN;",";TP;",";OO
840 FORX=1TONN:PRINT#1,BB$(X):NEXTX
850 CLOSE:RETURN
860 REM - READ SUBROUTINE -
870 CLS:PRINT@18,"READ LETTER FROM DISK":PRINT""
880 PRINT"ENTER NAME OF LETTER ";:O=8:I=0:GOSUB1040
890 NA$=B$(I)+"/TXT"
900 PRINT"":PRINT"ON WHICH DRIVE IS THE FILE (0-3)?";:
910 GOSUB1280:NA$=NA$+":"+A$
920 PRINT"":PRINT"THE FILE NAME IS ";NA$:PRINT"":PRINT"WHEN READY PRESS >ENTER<";:GOSUB1280:
    PRINT"":PRINT"**  READING DISK  **"
930 IFA$="E"THENRETURN
940 OPEN"I",1,NA$:INPUT#1,PP,NN,TP,OO
950 FORX=1TONN:INPUT#1,BB$(X):NEXTX
960 CLOSE:GOSUB1140:RETURN
970 REM - SINGLE KEY ENTRY -
980 A$=""
990 PRINT@P,CHR$(143);:L=1
1000 A$=INKEY$:IFA$<>""THENPRINT@P,A$;:RETURN
1010 L=L+1:IFL<16THEN1000
1020 PRINT@P," ";:FORL=1TO10:NEXT:GOTO990
1030 REM - MULT KEY ENTRY -
1040 PRINTCHR$(14);:B$(I)=""
1050 A$=INKEY$:IFA$=""THEN1050
1060 IFFL=1THENIFASC(A$)=44THEN1050
1070 IFA$=CHR$(13)THENPRINTCHR$(15):RETURN
1080 IFA$=CHR$(8)THENIFB$(I)<>""THENPRINTA$;:B$(I)=LEFT$(B$(I),LEN(B$(I))-1):
     GOTO1050:ELSEGOTO1050
1090 IFA$=CHR$(24)THENPRINTCHR$(29);STRING$(P,CHR$(25));CHR$(30);:GOTO1040
1100 IFASC(A$)<32ORASC(A$)>90THEN1050
1110 IFLEN(B$(I))=0THEN1050
1120 PRINTA$;:B$(I)=B$(I)+A$:GOTO1050
1130 REM - TYPE SUBROUTINE -
1140 IFOO<1ORNN<1THENRETURN
1150 CLS:PRINT"THE LETTER NOW READS:"
1160 FORII=1TONN:PRINTUSING"###. ";II;:PRINT" ";BB$(II):FORX=1TO250:NEXTX:NEXTII:PRINT"":
     PRINT"PRESS ENTER TO CONTINUE:";:GOSUB1280:RETURN
1170 REM - PRINT SUBROUTINE -
1180 IFNN<1THENRETURN
1190 CLS:PRINT"THE PRESENT TAB SETTING IS ";TA;", IS THIS OK? (Y/N)";
1200 GOSUB1280:IFA$="Y"THEN1230
1210 PRINT"":PRINT"WHAT IS THE TAB POSITION?";:O=2:I=0
1220 GOSUB1040:IFVAL(B$(0))<0ORVAL(B$(0))>64THENPRINT"?":GOTO1220:ELSETA=VAL(B$(0))
1230 PRINT"":PRINT"PRESS ENTER TO CONTINUE, E TO EXIT";:GOSUB1280:PRINT"":IFA$="E"THEN RETURN
1240 POKE16424,PP:IFPEEK(14312)>127THENPRINT"** PRINTER NOT READY **":GOTO1230
1250 LPRINTSTRING$(TP,138)
1260 OP=PP-(TP+NN)
1270 FORII=1TONN:LPRINTTAB(TA)BB$(II):NEXTII:LPRINTSTRING$(OP,138):RETURN
1280 GOSUB1300:GOSUB980:RETURN
1290 REM - LOCATE CURSOR -
1300 P=(256*PEEK(16417)+PEEK(16416))-15359:RETURN
1310 IFNN<1THENRETURNELSECLS:PRINT"ENTER ADDITIONAL LINES:":
     II=NN:O=OO:I=0:GOSUB390:RETURN
```

TRS-80™ USERS STOP!!!

WE'VE DONE IT AGAIN! MORE QUALITY ACCESSORIES STARTING WITH *TSHORT+* ! LVII & DOS SUPER SHORTHAND.

NEW!

NOW! WEB'S FAMOUS TSHORT™ EXPANDS!

SAVE MORE TIME THAN EVER... TYPING PROGRAMS WITH TSHORT+™!

TRS-80 Keyboard with decals installed.

Look at these new Features:

- NOW! 41 preprogrammed LV II and DOS statement keys.
- NOW! 11 "SAVEABLE" KUSTOM keys: Ten 10-character and one 64-character.
- NOW! Includes automatic keyboard DEBOUNCE.
- NOW! RELOCATABLE. Less than 1 K bytes of machine language in low or high memory.
- NOW! Available on formatted DISKETTES for 2 or more drives. Comes on cassette for LV II and single disk drives.
- Hold "SHIFT" or "CLEAR" and press desired key — entire statement is typed on screen. Installed "CRTL" key can substitute for "SHIFT"
- Complete decal set (see picture) included for both LV II and DOS systems.
- Features self-entering commands, i.e. CONT; GOTO10; KUSTOM.
- Automatic Close Paren. ")" may be programmed in KUSTOM.

TSHORT+ cassette
w/ instruction manual .... **$ 19.95**

TSHORT+ formatted
DISK version + manual .... **$ 24.95**

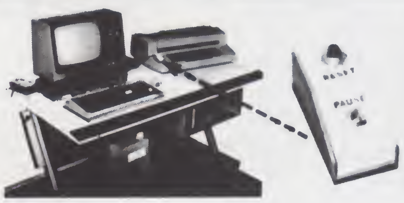
NEW!

TMEM™

Now, the best of both worlds! A battery supported memory giving you READ/WRITE capability with ROM security!

- Use for retaining important totals/variables, utility/system software (i.e. TSHORT, TLEC), development programs, monitors, etc.
- User programmable. Write enable switch + OUT statement ensures memory security.
- External plug-in module. Available in either 1K or 2K.
- Retains memory contents on power down or failure for 2 weeks or more. Built-in Batteries recharge automatically.
- Instant access to memory contents upon power up.
- Uses unassigned area of memory. No conflicts with other operating software.
- Compatible with either LV II or Disk Operating System. (Special cable required for LV II)

TMEM w/1K **$124.95**
or
TMEM w/2K **$174.95**

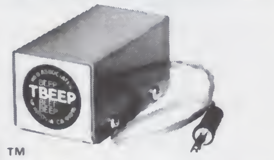


TSEL™

We'll convert YOUR IBM Selectric I or II to a high quality letter printer, totally compatible with your TRS-80.

- μp control — 512 character buffer.
- Special TRS-80 cable with custom "Pause", "Reset" switches. Our optional "Y" cable allows it to be plugged in with other printers.
- A superior word processing system.
- Compatible with Electric Pencil™ or Radio Shack's Scripsit™. Patches or modifications not required.
- Completely tested and ready to LPRINT.

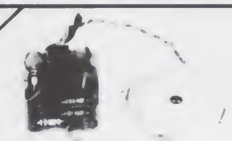
TSEL (cleaning and minor
service included) **$795.00**
(Options and shipping extra — call or write for special shipping instructions).



TBEEP 1™

For Level II and Disk Users — A self-contained audio alert beeper with a pager-like tone. Plugs in-line with "AUX" cable from your TRS-80 (Requires 9V Battery) ...... **$ 19.95**

NEW!



TBEEP 2™

A TBEEP kit you install inside your TRS-80 keyboard.

- Completely assembled and tested.
- Attach to keyboard's PC board with double-sided foam tape (included) and solder three wires to easily located points on keyboard unit. Installs in minutes.
- Uses power from your TRS-80. No battery required.
- **DEALERS!** Install when modifying your customer's TRS-80, and include TBEEP's simple four word BASIC command in your off the shelf Software.

TBEEP 2 Kit (with complete
instructions) .... **$ 12.95**

NEW!

TLEC™

Kit interfaces your IBM Electronic 50/60 to your TRS-80

- Kit includes Cable with Custom control switches for "Pause" and "Reset" and cassette with operating Software.
- Installs in minutes. Plug it in, no wire to cut or solder. Does not disturb IBM warranty.
- Needs no modifications or patches to popular word processing software.
- Requires either 1K TMEM or 2K TMEM. Load only once and forget it.

TLEC w/1K "TMEM" (with complete
instructions) **$239.95**
TLEC w/2K "TMEM" (with complete
instructions) **$289.95**

TPAK™

The best cassette tapes money can buy — AGFA 611. We use them for our production software, such as TSHORT™ Pack of 10 C-10 blank tape cassettes, boxes, and blank labels .... **$ 12.95**

TBASE™/TCHAIN™

2 powerful winners coming up. Watch our ads for release.

TBUFF™— OLD FAITHFULL

Stop your cassette when it should with our TBUFF cassette interface buffer.

(Specify recorder make & model) ... **$ 9.95**

Telephone Orders:
(714) 559-6249

PLEASE NOTE: All WEB Associates' products are designed to work with Level II **and** Disk Operating Systems unless specified otherwise.

Send Check or Money Order to:

WEB ASSOCIATES ✓28
P.O. Box 60 KJ, Monrovia, CA 91016
(California Residents add 6% tax)

DEALER INQUIRIES INVITED

UPS/C.O.D. — ADD $3.00

FOREIGN ORDERS:
Add 20% ($10 maximum)

Save more than 20%!

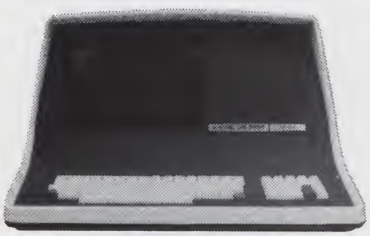
AMERICAN SQUARE COMPUTERS

NORTH STAR--INTERTUBE
THINKER TOYS--MICROTEK

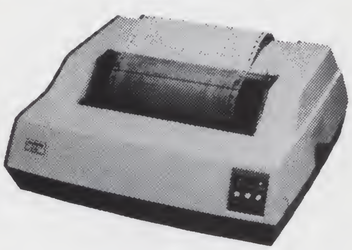
The smartest computers at the smartest price.

HORIZON QUAD & DOUBLE DENSITY

	LIST	ONLY
HORIZON-1-32K-D KIT	$1999	$1585
HORIZON-2-32K-D KIT	$2399	$1905
ASSEMBLED & TESTED	$2765	$2195
HORIZON-2-32K KIT QUAD	$2799	$2225
ASSEMBLED & TESTED	$3215	$2555
PASCAL FOR NORTH STAR ON DISK		49
POWERFUL NORTH STAR BASIC		FREE
TEI PT 212 COMPUTER 5 MHz	8000	6250
THINKER TOYS DISCUS/2D A&T	1149	949
DISCUS/2+2 1.2 MEGABYTES A&T	1549	1299
MEASUREMENT SYSTEM MEMORY A&T 4 MHz 64K		640
GODBOUT MEMORY		CALL FOR PRICE



INTERTUBE II SMART TERMINAL	$995	$745



MICROTEK PRINTER	750	675
ANADEX PRINTER	995	875
FLORIDA DATA PRINTER 600 CPS	4300	CALL FOR PRICE

Super Software at Reasonable Prices

MARYELLEN WORD PROCESSOR YOUR BEST BUY $38
TEXTWRITER III $125
EZ-80 TUTORIAL—LEARN MACHINE LANGUAGE $25
PDS FOR NORTH STAR—BETTER THAN CP/M $99
COMPILER FOR HORIZON—SECRET SUPERFAST CODE $100

10% Off Software Prices with Computers

VERBATIM THE BEST DISKETTES BOX OF 10 $29
WHICH COMPUTERS ARE BEST? BROCHURE...... FREE
NORTH STAR DOCUMENTATION, REFUNDABLE WITH COMPUTER $20

AMERICAN SQUARE COMPUTERS
✓56

KIVETT DR
JAMESTOWN NC
27282
(919) 883-1105

FORTH is an advanced language/ system for advanced programmers. MMSFORTH is a professional version tailored to the Radio Shack TRS-80 Model I.

microFORTH PRIMER (required) $15.00
MMSFORTH System Diskette (1 drive & 16 K req.) $64.95
or MMSFORTH System Cassette (Level 2, 16K) $44.95
Shipping $2, Mass. orders add 5% tax.

Interpreter, Compiler, Assembler, Full-Screen Editor
Structured & Modular Programming
Expandable Instruction Set w/Source Code
Graphics, Strings, Arrays, Double-Precision
Very Fast and Compact, Virtual Memory
Includes 5 Demo Programs w/Source Code

mms FORTH

MILLER MICROCOMPUTER SERVICES
61 Lake Shore Road, Natick MA 01760 (617) 653-6136
Send SASE for free information.
✓125

OHIO SCIENTIFIC
USERS!

Read

PEEK (65)
The Unofficial OSI Users Journal

- User Equipment Reviews
- Hardware Mods
- Software Exchange
- Peeks & Pokes
- Bugs & Fixes

Send $8.00 for 12 issues to: ✓178

PEEK (65)
62 Southgate Ave.
Annapolis, MD 21401

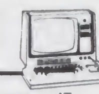
Apple II 16K Plus	$995.00
Apple Disk w/Controller	529.00
Apple Disk w/o Controller	475.00
Apple Integer Card	180.00
Pascal Language SYS	450.00
High Quality 16K RAM Set	89.00
BPI General Ledger SYS	300.00
Plan Modeling System	95.00
MNTN. HDWR. Clock Board	175.00
Trendcom 100 Printer	375.00
Trendcom 200 Printer	595.00
Centronics 779 Printer w/Int. Card	1395.00
Graphics Tablet	749.00
Apple Writer Program	69.00
D.C. Hayes Micromodem II	349.00

OHIO SCIENTIFIC

CI-P Personal Computer	$329.00
Morse Code Transceive Program for Challenger CI-P Computer Send & Receive CW	14.95
Space Shuttle Simulation Program for C1-P Computer	14.95

OLENSKY BROS., INC.
COMPUTER SALES DIVISION
3763 Airport Blvd. • Mobile, AL 36608
(205) 344-7448
✓130

VISA WE PAY SHIPPING master charge



Bluebird's Inc. ✓175

2267 23rd STREET • WYANDOTTE, MI. 48192 • 313-285-4455

SOFTWARE for the TRS-80™

Compress-IT.......... REQUIRES DISK BASIC & ANY RAM SIZE $24.95
REMOVE UNNECESSARY SPACES & REMARK STATEMENTS IN A PROGRAM COMPRESSES A PROGRAM INTO MAXIMUM MULTIPLE STATEMENT LINES REDUCE PROGRAM RAM REQUIREMENTS TO A MINIMUM CREATE FASTER RUN-TIME PROGRAMS FROM A WELL DOCUMENTED SOURCE PROGRAM COMPRESS-IT CONSISTS OF 2 PROGRAMS

Simplify-IT.......... REQUIRES DISK BASIC & ANY RAM SIZE $24.95
BREAK DOWN A COMPLEX PROGRAM & LIST PROGRAM IN SINGLE STATEMENT LINES TITLES & NUMBERS EACH PAGE IN 11 INCH FORMS LENGTH ELIMINATES CHOPPED OFF LINES INHERENT WITH SOME PRINTERS SEARCHES & FINDS REQUESTED TEXT OR COMMANDS-STATEMENTS-ETC LISTS ALL VARIABLES AND DEFINITIONS USED IN A PROGRAM WORKS WITH OR WITHOUT A LINE-PRINTER SIMPLIFY-IT CONSISTS OF 3 PROGRAMS

TEXT.......... REQUIRES DISK BASIC & 32K MIN RAM $49.95
A COMBINATION TEXT PROCESSOR & ELECTRIC FILING CABINET EXECUTIVE MAINTAINS DIRECTORY FILES AND PREVENTS FILE DESTRUCTION EVEN IF THE TEXT-FILE IS GIVEN THE SAME NAME AS A EXISTING FILE AUTOMATIC BACKUP OPTION EXTENSIVE EDIT & TEXT BLOCK MOVE OPTIONS FAST SEARCHES THROUGH EVERY FILE EVER WRITTEN WITH PROVISION FOR CHANGING DISKS DURING SEARCH FORMATS & PRINTS TEXT FILES TO USER SPECIFICATIONS WORKS WITH 1 TO 4 DISK DRIVES

List 'n File: Names & Things REQUIRES DISK BASIC & 32K MIN RAM $34.95
CREATE & MAINTAIN MULTIPLE CATEGORY FILES CATEGORY FILES ARE USER DEFINED AND AUTOMATICALLY MAINTAINED ALLOWS ANY KIND OF INFORMATION TO BE EASILY FILED EXTENSIVE FILE EDITING CAPABILITIES EXTENSIVE FILE SEARCH OPTIONS PRINTS LABELS OR DATA LISTINGS OF ALL FILES OR SEARCHES-FOR AND PRINTS ONLY FILES CONTAINING USER SPECIFIED KEY WORDS

Compare.......... REQUIRES DISK BASIC & ANY RAM SIZE $14.95
COMPARE ANY TWO PROGRAMS UTILITY LISTS DIFFERENCES IN THE PROGRAMS TO VIDEO AND/OR LINE-PRINTER A VERY USEFUL TOOL TO EVALUATE FREQUENTLY UPDATED PROGRAMS

DISK.......... REQUIRES DISK BASIC & 32K MIN RAM $14.95
THE HOW-TO RUDEMENTS OF YOUR DISK SYSTEM INTERACTIVE WITH LOTS OF DYNAMIC EXAMPLES HELPS A BEGINNER USE THE DISK SYSTEM RECOMMENDED FOR USE IN SCHOOLS

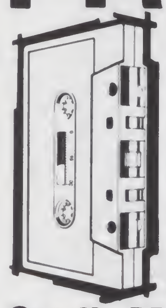
Math/Stat Pac....... LEVEL 2 TAPE OR DISK SYSTEM & ANY RAM SIZE $39.95
LINEAR · MULTIPLE · POLYNOMIAL · GEOMETRICAL · EXPONENTIAL REGRESSION ANALYSIS SIMULTANEOUS EQUATIONS BARGRAPH WITH AUTO-AXIS PLOT ANY EQUATION OVER ANY RANGE (REDUCE OR ENLARGE) WITH AUTO-AXIS STATDAT DATA BASE ALLOW ENTRY AND EDITING OF DATA WHICH CAN BE STORED ON TAPE OR DISK STATDAT FILES ARE READ BY THE ANALYSIS PROGRAMS A PACKAGE OF INTERESTING EQUATIONS IS INCLUDED FOR USE BY THE GRAPHIC PROGRAMS MATH/STAT PAC CONSISTS OF 9 PROGRAMS

ALL BLUEBIRD'S PROGRAMS ARE IMMEDIATELY AVAILABLE, AND ARE SHIPPED WITHIN 1-WEEK OF RECEIPT PROGRAMS WILL BE SHIPPED ON TAPE CASSETTE UNLESS YOU REQUEST A DISK COPY NOTE PROGRAMS ARE NOT SUPPLIED ON A DISKETTE WITH TRS-DOS DISK COPIES ARE ON DATA-DISKETTE ONLY CREDIT CARD ORDERS MAY BE PLACED BY PHONE (313) 285-4455

** BLUEBIRD'S NEW FAST SERVICE ADDRESS **

VISA 2267 23rd STREET
WYANDOTTE, MI. 48192 master charge

FOR MINI-COMPUTERS
FREE Special Trial Offer!!



When you order 23 C-10 cassettes featuring Precision transport mechanism and Digital pressure pad. (In individual, dust-proof, plastic boxes with labels) for a low introductory price of $19.95, we will send you FREE 1 C-10 cassette for you to test. If you are not delighted with its performance you can return the unused 23 units for a full refund. Please add $1.05 to cover postage and handling. N.Y. Res. add 7%. VISA and Master Charge accepted.

Write for pricing on larger quantities.

Studio Magnetics Co., Inc.
✓179

SM
83 Carlough Road
Bohemia, N.Y. 11716
516-589-4300

The Human Adventure is just beginning . . .

. . . with Med System's exciting new simulation. Imagine yourself reduced in size and injected into a human body. The entire cardiovascular system becomes your highway. Your computer becomes your eyes and ears, allowing you to explore all major organ systems. But beware! The body deals with invading organisms mercilessly.

Combine adventure with education! The Human Adventure teaches anatomy and function to anyone who can read.

Med System's Human Adventure is a 16K TRS-80 program requiring Level II for loading. It is shipped with a two week money back guarantee and an extra blank C-10 cassette for back-up copying.

TRS-80 16K cassette $9.95 ✓129
N.C. residents please add 4% tax.

Med Systems Software
P.O. Box 2674, Chapel Hill, N.C. 27514

An Extraordinary Offer to introduce you to the benefits of Membership in

ELECTRONICS BOOK CLUB

invites you to take this 1,302-page robotics library for only $1.99

You Get ALL FOUR Of These Huge Books For Only $1.99

- ✓ Only $1.99 for ALL FOUR!
- ✓ Regular List Price $47.80
- ✓ Top-Quality Hardbinding
- ✓ Contains the very latest info on Robotics!
- ✓ Almost 800 illustrations
- ✓ Contains over 500,000 words
- ✓ 1,302 data-packed pages

Build Your Own Working Robot

Complete instructions—plans, schematics, logic circuits, and wiring diagrams—for building Buster, the most unique pet in the world! Not for novices, Buster is a sophisticated experiment in cybernetics. You build him in 3 phases, and watch his personality develop as you add progressively more advanced circuitry to his mainframe. The first-phase robot, Buster I, is "leash-led," and dependent on his master for decision-making; Buster II has a basic brain; equipped with a wireless mike, he can enter a room and talk with the occupants. Buster III responds when called, and when "hungry" finds his charger, and plugs himself in. Watch his personality evolve as you build him from the ground up in a learning experience unparalleled in electronic construction. 238 pps., 117 illus. List $8.95.

Handbook Of Remote Control & Automation Techniques

A practical, step-by-step guide to designing, building, and installing hundreds of remote control systems, and scores of automated devices...from garage door openers to light sensors, from intercom controls to electromechanical timers—to interfacing a microprocessor with household devices. You'll learn how to apply electronic and mechanical techniques to remote-control with computers, with audible tones, with ultrasonics, with radio waves, with light beams, with dozens of special systems. You can build light and power failure sensors, position indicators, tone-operated systems, tone generators, RC hydraulic devices...and you'll see how to interface mechanical devices, hydraulic systems, and electric motors with electronic systems. 294 pps., 250 illus. List $12.95.

Digital Interfacing With An Analog World

A GIANT 406-page handbook that shows you how to design circuits to interface microprocessors, computers, telephones, and other digital devices with the analog world...that shows you how to *really* put your microcomputer to work to measure certain conditions, or to control external devices. Tells you all about how to go about it—how to convert energy produced by pressure, force, position, temperature, etc. into an electrical voltage or current your microcomputer can deal with. It shows you, tells you, describes and discusses things you can do with those I/O ports other than connect them up to a prefabricated peripheral! It's a "meaty" volume chock-full of practical info on a wide range of topics for engineers, computer hobbyists, engineering technicians, and robotics builders. 406 pps., 277 illus. List $12.95.

The Complete Handbook Of Robotics

How to design and build ANY kind of robot...including ones with microprocessor "brains"—PLUS how to interface robots with computers! It's a single sourcebook that contains all the techniques you'll need for creating, designing, building, and operating your own robot from beginning to end...with enough options to create a whole family of robotic wonders—controls can be electrical or electronic; power can be electrical, hydraulic, or pneumatic; your robot can operate by radio control or with a full range of sensors to move about on its own. This practical volume gives ALL the info needed to build a walking, talking friend and companion, or even a helpful servant. Includes 7 Chapters on advanced robot circuits, controls, and sensors. 364 pps., 137 illus. List $12.95.

Let us send you this 4-volume, 1,302-page Robotics Library as part of an unusual offer of a Trial Membership in Electronics Book Club.

Here are quality hardbound volumes, each especially designed to help you increase your know-how, earning power, and enjoyment of electronics. Whatever your interest in electronics, you'll find Electronics Book Club offers practical, quality books that you can put to immediate use and benefit.

This extraordinary offer is intended to prove to you through your own experience, that these very real advantages can be yours...that it *is* possible to keep up with the literature published in your areas of interest, and to save substantially while so doing. As part of your Trial Membership, you need purchase as few as four books during the coming 12 months. You would probably buy at least this many anyway, without the substantial savings offered through Club Membership.

To start your Membership on these attractive terms, simply fill out and mail the coupon today. You will receive the 4-volume Robotics Library for 10-day inspection. YOU NEED SEND NO MONEY. If you're not delighted, return the books within 10 days and your Trial Membership will be cancelled without cost or obligation.

ELECTRONICS BOOK CLUB, Blue Ridge Summit, Pa. 17214

Facts About Club Membership

- The 4 introductory books carry a publisher's retail price of $47.80. They are yours for only $1.99 for all 4 (plus postage/handling) with your Trial Membership.
- You will receive the Club **News**, describing the current Selection, Alternates, and other books, every 4 weeks (13× a year).
- If you want the Selection, do nothing; it will be sent to you automatically. If you do not wish to receive the Selection, or if you want to order one of the many Alternates offered, you simply give instructions on the reply form (and in the envelope) provided, and return it to us by the date specified. This date allows you at least 10 days in which to return the form. If, because of late mail delivery, you do not have 10 days to make a decision and so receive an unwanted Selection, you may return it at Club expense.
- To complete your Trial Membership, you need buy only four additional monthly Selections or Alternates during the next 12 months. You may cancel your Membership any time after you purchase these four books.
- All books—including the Introductory Offer—are fully returnable after 10 days if you're not completely satisfied.
- All books are offered at low Member prices, plus a small postage and handling charge.
- **Continuing Bonus:** If you continue after this Trial Membership, you will earn a Dividend Certificate for every book you purchase. Three Certificates plus payment of the nominal sum of $1.99 will entitle you to a valuable Book Dividend of your choice which you may choose from a list provided Members.

ELECTRONICS BOOK CLUB
Blue Ridge Summit, Pa. 17214

Please open my Trial Membership in ELECTRONICS BOOK CLUB and send my 4-volume Robotics Library, invoicing me for only $1.99 plus shipping. If not delighted, I may return the books within 10 days and owe nothing, and have my Trial Membership cancelled. I agree to purchase at least four additional books during the next 12 months after which I may cancel my membership at any time.

Name ____________________ Phone __________

Address ______________________________

City ______________________________

State ______________ Zip __________

(Valid for new Members only. Foreign and Canada add 15%.) MC-480

David Barnhart WB7OBG
4432 West Ironwood Dr.
Glendale AZ 85302

A Better Troubleshooting Aid

This technique allows you to find problems in a computer that is functionally "shot."

There have been several articles in *Kilobaud Microcomputing* about troubleshooting microcomputers. I have noticed that a very simple troubleshooting aid has never been mentioned. Most troubleshooting techniques require the system to at least perform some minimal function, and even a front panel unit must have the ability to jam an instruction into the MPU. This article describes a device that is useful in troubleshooting a microcomputer that is completely dead or, upon power up, just "goes crazy."

I am a computer technician by trade, doing troubleshooting and debugging on microprocessor-controlled logic circuits. The processor I deal with is the Motorola 6800. Usually, a board comes directly from the manufacturing department to our quality control department with little touch-up. Applying power to the board for the first time usually opens up a Pandora's box of solder shorts and, to a lesser extent, defective components. Even an excellent scope is of little help identifying shorted or open address or data lines under these conditions, much less the more subtle problems.

We have been able to cut our troubleshooting time by as much as two-thirds using the device in this article. All that is required is this plug-in circuit and either an oscilloscope or a logic probe that has the ability to detect pulses and clock signals as well as the high and low logic levels.

The first step is to remove the resident MPU chip from its socket on the CPU board. This device is simply plugged in place of the resident MPU, via 40-pin DIP header and ribbon cable (see Fig. 1). At the other end of the cable is, basically, another MPU chip wired such that only its timing, power and ground signals are connected to the resident CPU board. The nonresident MPU's address pins are left unconnected, and its data lines are hard-wired so that it always sees a NOP (no operation) instruction. For the 6800, this is hex 01.

The address lines from the resident CPU board's MPU socket are brought, via the DIP header and ribbon cable, to DIP switches, so that each address line can individually be set at a high or low (one or zero) level. The resident data bus is brought out to wire-wrap pins to facilitate checking logic levels (see Fig. 2).

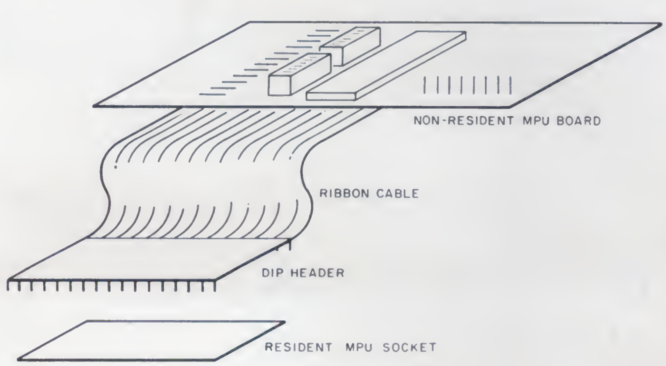


Fig. 1.

Construction

Construction is simple and to the point. You can use perfboard and point-to-point wiring. No specific resistor values used in the switches are required. Try to keep the ribbon cable as short as practical, as it is susceptible to noise. Mine is about 18 inches long and works fine.

Be sure you thoroughly understand how the ribbon cable connects to the DIP header. Opposite pins on the header will be adjacent conductors on the ribbon cable, which means that the conductors will be in non-sequential order. As an alternative to physically checking the data bus at the wire-wrap pins with a scope or logic probe, you can use one of the two circuits shown in Figs. 3 and 4. Each will allow you to put an LED on each data line so that the data on the bus can simply be checked visually.

Please note that the pin-outs and the op-code for the NOP apply to the 6800 MPU only. For any other processor, the pin-outs and the hard-wired 01 for the NOP must be changed accordingly.

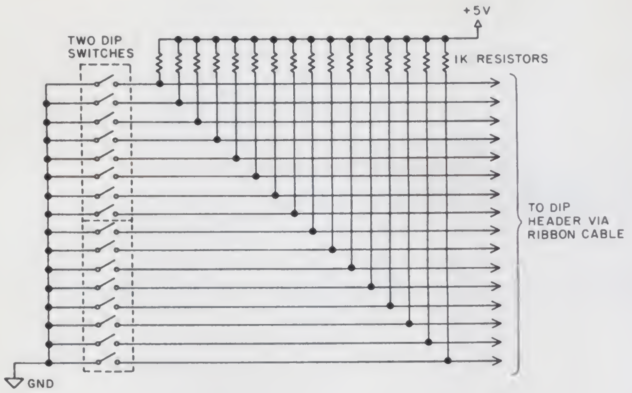
It is advisable to cover the back of the board with a protective material such as nonconductive foam, as the wiring is fragile. It is also advisable to plug this circuit in sometime before a system failure to ensure that it works correctly.

How It Works

With the nonresident MPU's data bus tied to look like a NOP instruction, this processor puts out an address on its address bus (which is not connected to anything, remember) and finds a NOP instruction. It then executes the NOP, increments the program counter by one and puts out the next address, which is the present address plus one.

With the processor always executing a NOP, then incrementing the address on the bus, you can see that it will continually cycle through all 64K addresses, doing a read on every one. But the address that the resident CPU board sees is tied high or low through the DIP switches. It thinks it is supposed to do a read of one address continually. Remember: The Read/Write line and the rest of the system timing signals are left connected (see Fig. 2).

Now, since we are doing a continuous read of a single address (which you have selected with the DIP switches), we can follow that address out through all the buffers, decoding and chip selects to see if the proper device is selected and read. I hope that, sometime prior to the system breakdown, you will have plugged in this circuit and addressed a few selected positions in ROM somewhere. This way you will have a short list of addresses and the data that should be found there.



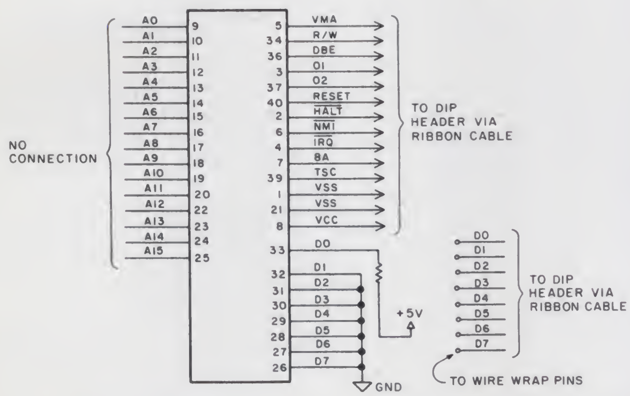


Fig. 2. Address, data and timing signal lines have the same pin-outs on the DIP header as they do on the MPU chip.

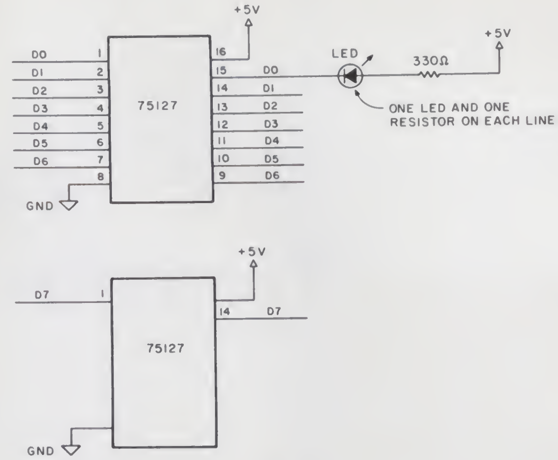


Fig. 3.

Normally, the troubleshooting procedure would occur as follows: After plugging in the nonresident processor board and powering up, check to see that all the system timing signals look OK. (You know what they look like because you did this once when everything was working.) Be sure to look at each signal on both sides of every buffer on that signal.

Next, using the DIP switches, bring up the address lines one at a time. With all other address lines low except the one you are checking, it will be easy to tell if any two are shorted somewhere. Normally, when two lines are shorted and one is low, it will try to pull the other line low.

If you do this every day on the same type of system, as I do, you will usually be able to tell the difference between a ground low, a TTL low and a shorted low. A ground low looks like the nearest ground run; a TTL low is not quite at ground potential; and a shorted low is generally a little above a TTL low.

Next, select an address in a ROM somewhere in your system, preferably the ROM in which your Reset-and-Go routine is located. Check to see that it is getting all its proper signals, such as chip select and Read/Write. Chip select is usually an active low signal, that is, the chip is selected when the chip select is low.

As always, have the schematics in front of you before attempting to do any serious troubleshooting. Many times a pair of the timing signals are ANDed in some obscure portion of the logic, but are vital to the system's operation. A good example is valid memory address (VMA) or read/write (R/W) being ANDed with phase two of the system clock, usually called TTL Phase Ø2, or Bus Phase Ø2.

Select some of the ROM addresses whose contents you have written down. If the data you get back is wrong, you are getting warm.

Write down the data in *binary*, not hex, as it should be, and below it as it appears. Put the ones and zeros in a line next to each other. By comparing the good and bad data, you should be able to determine which line or lines have problems. For example, if, when you read a location in ROM, all of the data bits except bit 4 compare with what you have read in the past, then you have a problem somewhere on the data bus with data bit 4. Two bits that go high or low together, but not independently, are shorted together. Remember also that the address or data bus lines may not be laid out on the board in sequential order. The even- and odd-numbered lines may be grouped together, for instance.

You should also select an address that is not used anywhere in your system and, one at a time, bring the data bus lines high to see if they go high at the nonresident board. This can be accomplished by touching each line with a resistor with its other end attached to +5 V (VCC). The ones that are not high should either be Tri-state or low.

Helpful Hints

Sometimes you may have to go through this procedure several times to uncover the problem. Be sure that you have checked those signals on both sides of every buffer. It is easy to miss one.

Do not be too quick to blame the MPU chip if it seems as though you have checked everything else. In all my experience I have only seen one MPU chip go bad. I do recommend using a spare MPU chip in the nonresident board. Then you can substitute at any time to reassure yourself.

Sometimes a technician's best friend is his X-acto knife. Frequently it is impossible to determine if the run is shorted or if the chip at the end of the run is shorted internally.

The easiest way to tell is to carefully cut the run at the base of the pin of the suspect IC. Whichever of the two is still shorted is the guilty party. Be judicious about cutting PC board runs, though, as you can get carried away and miss one of your cuts. Always remember to repair any cuts you have made or you will compound the problem at hand.

Conclusion

I have used this circuit and accompanying technique several times a week for about three months, and they have proven foolproof. If the average hobbyist, who may have very little experience along these lines, has any questions, just drop me a line with an enclosed SASE for your reply. ■

Acknowledgements

I must extend my appreciation to Joe Bencze, who came up with the idea, and also to Robert Bennet and John Gatti.

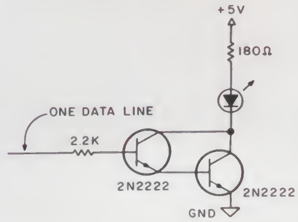


Fig. 4.

✓Reader Service—see page 257

Interfacing a Diablo HyType I to an SWTP 6800

Provided here are both the hardware and support software for this project.

Phil Hughes
PO Box 2847
Olympia WA 98507

This article describes how I interfaced a Diablo HyType I print mechanism to a Southwest Technical Products 6800 computer system. I've described both the hardware for a parallel interface and the support software for what could be called teletypewriter simulation. Also included are software interfaces to Technical Systems Consultants FLEX 2.0 and Mini-FLEX operating systems.

I chose the Diablo Model 1200 HyType I Printer because of its high print quality, moderate speed and high reliability, and because reconditioned mechanisms have become available at reasonable costs.

For those who are not familiar with the HyType I, it uses a plastic disk (called a daisy wheel) to form letter-quality characters. Like the carriage and platen, the daisy wheel is positioned by a servomechanism. This design offers print quality equal to a finely tuned IBM Selectric with only a few moving parts. Table 1 shows the characteristics of the printer.

All the logic for the printer mechanism is contained on three circuit boards mounted on the print mechanism chassis. This includes a parallel interface unique to daisy wheel printers. The additional hardware required to get the printer operating consists of a power supply capable of +5 V dc at 4 A, +15 V dc at 9 A peak and −15 V dc at 9 A peak (100 W is the typical average) and circuitry necessary to connect the printer parallel interface to an SWTP input/output port.

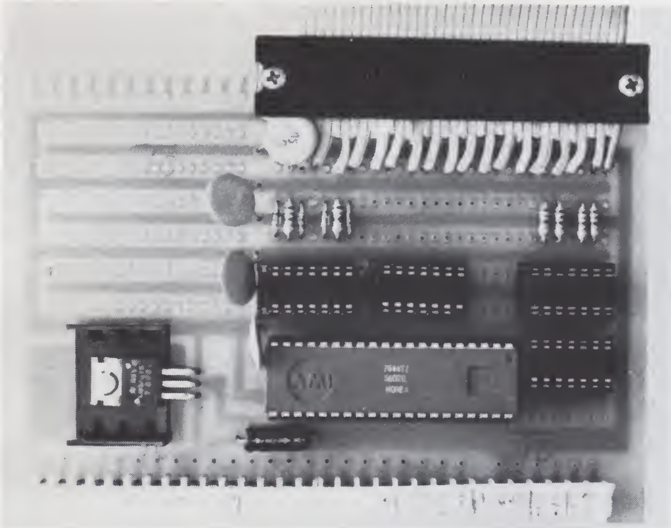
I purchased my power supply (a surplus commercial unit) and the interface and power cables. For those who choose to make their own cables, the mating power connector is a Winchester MRAC14SJTCH13 *with* Winchester 100-0919 socket (female) contacts. The mating input/output connector is a Winchester MRAC34JTDH with Winchester 100-092S socket (female) contacts. Table 2 identifies the contact usage of the power and I/O connectors.

SWTP system with Diablo HyType printer.

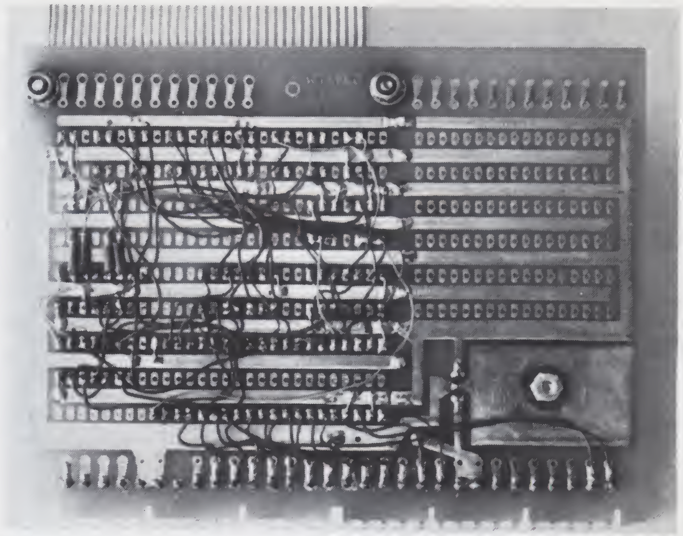
I/O Line Signals

Before I describe the I/O interface, I will describe the signals that appear on the Diablo interface connector. A true logic level is defined as 0 volts, and a false logic level is defined as +5 volts. The input lines are as follows:

SELECT PRINTER—All input lines except SELECT READY are inhibited until this signal is true.



Interface prototyping board (top view) designed to fit an SWTP I/O slot.



Interface board (bottom view).

SELECT READY—Enables the three output status lines: CHARACTER READY, CARRIAGE READY and PAPER FEED READY.

DATA—There are eleven data lines that receive binary-coded information representing an ASCII character, a carriage movement command or a paper feed command. When representing an ASCII character, only the low-order seven bits are used. When representing a carriage movement command, the ten low-order bits designate the distance the carriage is to be moved in multiples of 1/60 inch. When representing a paper feed command, the ten low-order bits designate the number of vertical 1/48 inch increments that the paper is to be moved. The high-order bit determines the direction.

CHARACTER STROBE—Used to load a 7-bit ASCII character code.

CARRIAGE MOTION STROBE—Used to load the 11-bit carriage movement command.

PAPER FEED STROBE—Used to load the 11-bit paper feed command.

RESTORE COMMAND—Causes the printer to perform a restore sequence, which consists of positioning the carriage to the left print column, synchronizing the print wheel with its logic and resetting the printer logic.

RIBBON LIFT COMMAND—Used to raise or lower the ribbon cartridge.

The output lines are as follows (note: all output lines are inhibited until SELECT PRINTER or SELECT READY signals are true):

PRINTER READY—Indicates the printer is properly supplied with power.

CHARACTER READY—Indicates the printer is ready to accept a character command.

PAPER FEED READY—Indicates the printer is ready to accept a paper feed command.

CARRIAGE READY—Indicates the printer is ready to accept a carriage command.

CHECK—Indicates that due to a machine, controller or power supply problem, a previously received carriage command has not been successfully completed.

PAPER OUT—Not supported by Diablo hardware.

I/O Interface

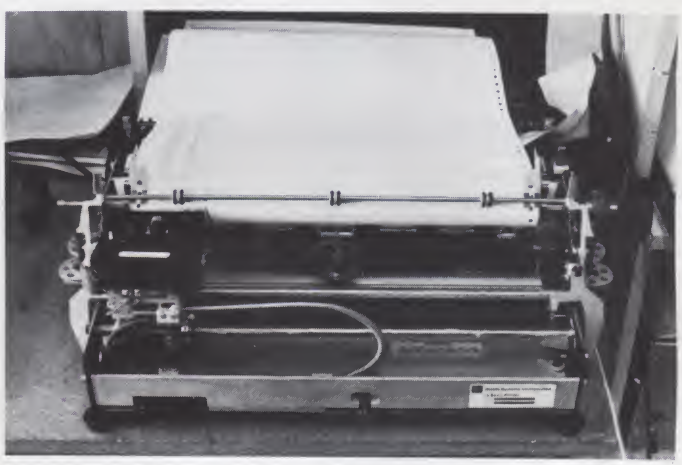
The interface diagrammed in Fig. 1 is built on a prototyping board designed to fit an SWTP I/O slot. The logic consists of a 6820 peripheral interface adapter (PIA), some NAND and NOR gates that are used to combine signals and some inverters used as line drivers. The NAND/NOR logic is needed so one PIA can handle all the required signals.

This configuration required sacrificing the capability of individually checking the ready status lines (CHARACTER, CARRIAGE, PAPER FEED and PRINTER). Also, the SELECT READY and SELECT PRINTER lines are wired true, thus losing the capability of individually addressing multiple printers on one interface.

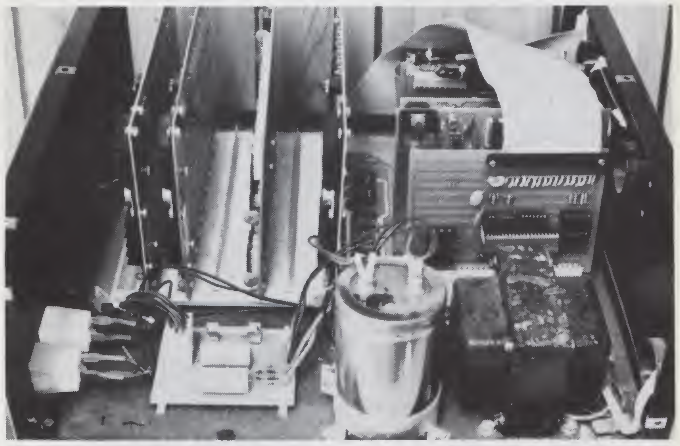
Three NAND gates and one

Print Speed	30 characters per second average
Character Set	96 characters
Print Line	132 characters at 10 cpi
Paper Width	15 inches maximum
Carriage Return Time	400 msec maximum
Tabulation	Right or left direct to column address
Column Spacing	60 positions per inch
Paper Feed	Up or down
Paper Feed Spacing	48 positions per inch
Paper Feed Speed	4 inches per second
Character Element	Interchangeable wheel
Ribbon	Cartridge: Cloth or film
Physical:	
Height	8.55 inches
Width	23.10 inches
Depth	13.11 inches
Weight	30 pounds

Table 1. HyType I characteristics.



Printing with the Diablo.



Interface installed in SWTP I/O slot 7.

Power:	
Signal	Pin
+ 15 High Current	E,H
+ 15 Low Current	F,K,D
– 15 High Current	M,P
– 15 Low Current	N,R
+ 5	J,L
Ground Return 15	A,B
Ground Return 5	C
I/O:	
Signal	Pin
Data 1	h
Data 2	j
Data 4	m
Data 8	f
Data 16	k
Data 32	i
Data 64	g
Data 128	d
Data 256	b
Data 512	V
Data 1024	F
Restore	E
Character Strobe	P
Carriage Strobe	K
Paper Feed Strobe	C
Select Printer	S
Select Ready	H
Ribbon Lift	M
Printer Ready	a
Check	B
Paper Out	R
Character Ready	Y
Carriage Ready	W
Paper Feed Ready	c
Ground	A,D,J,N,T,U,X

Table 2. Connector pin usage.

NOR gate (used as an inverter) are used to "AND" the CB2 signal, which is used as the load strobe, with the appropriate strobe select from PIA data lines PA4, PA5 or PA6. Two NOR gates and a NAND gate are used to "AND" the four ready lines from the printer, which are then fed into the PA7 line of the PIA.

Lines PA7, CA1 and CB1 are used as inputs. All other PIA lines (PA0–PA6, PB0–PB7, CA2 and CB2) are used as outputs. Line CB1 is designed to be a manual interrupt input.

Software

The software portion of the interface consists of a port initialization routine and a printer driver. Within the printer driver are routines to handle control characters.

The initialization routine (HINIT) is called before printing is attempted to set up both the PIA on the interface board within the computer and to issue a restore command to the printer. Setting up the PIA consists of selecting the data direction registers, loading them so that lines PA0–PA6 and PB0–PB7 are set up as outputs and line PA7 is set up as an input.

The PIA control register is then set to select the data registers, set up the appropriate handshaking for the CA1, CB1 and CB2 lines and lift the ribbon

Listing 1. HYTEST—Diablo test routine.

```
                    * HYTEST 6-27-79 @ 1810
                    * EQUs
0003                VERSION EQU    3          PROGRAM VERSION NUMBER
7103                WARMS   EQU    $7103      MONITOR RE-ENTRY ADDRESS
E1AC                INEEE   EQU    $E1AC      MONITOR GET CHARACTER
E1D1                OUTEEE  EQU    $E1D1      MONITOR PUT CHARACTER
0018                TERMC   EQU    $18        TERMINATE CHARACTER (CTL X)
E07E                PDATA1  EQU    $E07E      MONITOR PUT STRING
0006                CHRMOV  EQU    6          1/10 inch char spacing
0008                LINMOV  EQU    8          6 lines per inch
0042                LPP     EQU    66         lines per page
801C                HPIA    EQU    $801C      PORT 7
003C                CRAMSK  EQU    %00111100  CONTROL REG A MASK
002C                CRBMSK  EQU    %00101100  CONTROL REG B MASK
B000                        ORG    $B000
B000 20 01                  BRA    HINIT      Humor VERSION
B002 03                     FCB    VERSION    PROGRAM VERSION NUMBER
B003                HINIT   EQU    *          PORT AND PRINTER INITIALIZATION
B003 CE 80 1C               LDX    #HPIA      GET PORT ADDRESS
                    * Setup data direction registers
B006 6F 01                  CLR    1,X        SELECT DDRA
B008 6F 03                  CLR    3,X        SELECT DDRB
B00A 86 7F                  LDA A  #$7F       1 IN, 7 OUT
B00C A7 00                  STA A  0,X        Set DDRA
B00E 86 FF                  LDA A  #$FF       8 OUTPUTS
B010 A7 02                  STA A  2,X        Set DDRB
B012 86 3C                  LDA A  #CRAMSK    SETUP CONTROL REGISTERS
B014 A7 01                  STA A  1,X        Set CRA
B016 86 2C                  LDA A  #CRBMSK
B018 A7 03                  STA A  3,X        Set CRB
B01A 86 08                  LDA A  #$08       RESTORE PRINTER
B01C A7 00                  STA A  0,X
B01E 6F 00                  CLR    0,X
B020 7F B0 F5               CLR    HCHCNT     Clear character
B023 7F B0 F6               CLR    HLNCNT     and line counters
B026 39                     RTS
B027                HOUT    EQU    *          OUTPUT CHARACTER ROUTINE
B027 FF B0 F3               STX    HSAVEX     SAVE USER X
B02A BD B0 95               JSR    HRDY       CHECK FOR READY
B02D 37                     PSH B
B02E 81 20                  CMP A  #$20       SPACE OR CONTROL?
B030 27 08                  BEQ    HSPACE     SPACE
B032 2D 16                  BLT    HCNTRL     CONTROL
B034 C6 10                  LDA B  #$10       SETUP CHAR STROBE FUNCTION
B036 E7 00                  STA B  0,X
B038 A7 02                  STA A  2,X        PRINT CHAR
B03A                HSPACE  EQU    *
B03A 7C B0 F5               INC    HCHCNT     INC CHAR COUNT
B03D C6 20                  LDA B  #$20       carrage move
B03F 86 06                  LDA A  #CHRMOV    1 character space
B041 E7 00                  STA B  0,X
B043 A7 02                  STA A  2,X
B045                HRET    EQU    *          RESTORE & RETURN
B045 33                     PUL B
B046 FE B0 F3               LDX    HSAVEX     RESTORE USER X
B049 39                     RTS
B04A                HCNTRL  EQU    *          HANDLE CONTROL CHARS
B04A 81 0D                  CMP A  #$D        CR?
B04C 27 0A                  BEQ    HCR
B04E 81 0A                  CMP A  #$A        LF?
B050 27 19                  BEQ    HLF
B052 81 0C                  CMP A  #$C        FF?
B054 27 2C                  BEQ    HFF
B056 20 ED                  BRA    HRET       forget it
B058 F6 B0 F5       HCR     LDA B  HCHCNT     get char count
B05B 86 06                  LDA A  #CHRMOV    char length
B05D BD B0 AA               JSR    MULT       figure distance
B060 8A 24                  ORA A  #$24       carrage strobe & direction
B062 A7 00                  STA A  0,X
B064 E7 02                  STA B  2,X
B066 7F B0 F5               CLR    HCHCNT     clear char coun
B069 20 DA                  BRA    HRET       return

B06B 86 40          HLF     LDA A  #$40       paper feed strobe
B06D A7 00                  STA A  0,X
B06F 86 08                  LDA A  #LINMOV    Increments/line
B071 A7 02                  STA A  2,X
B073 B6 B0 F6               LDA A  HLNCNT     GET LINE COUNT
B076 4C                     INC A             INCREMENT
B077 B1 B0 D6               CMP A  HLPPCT
```

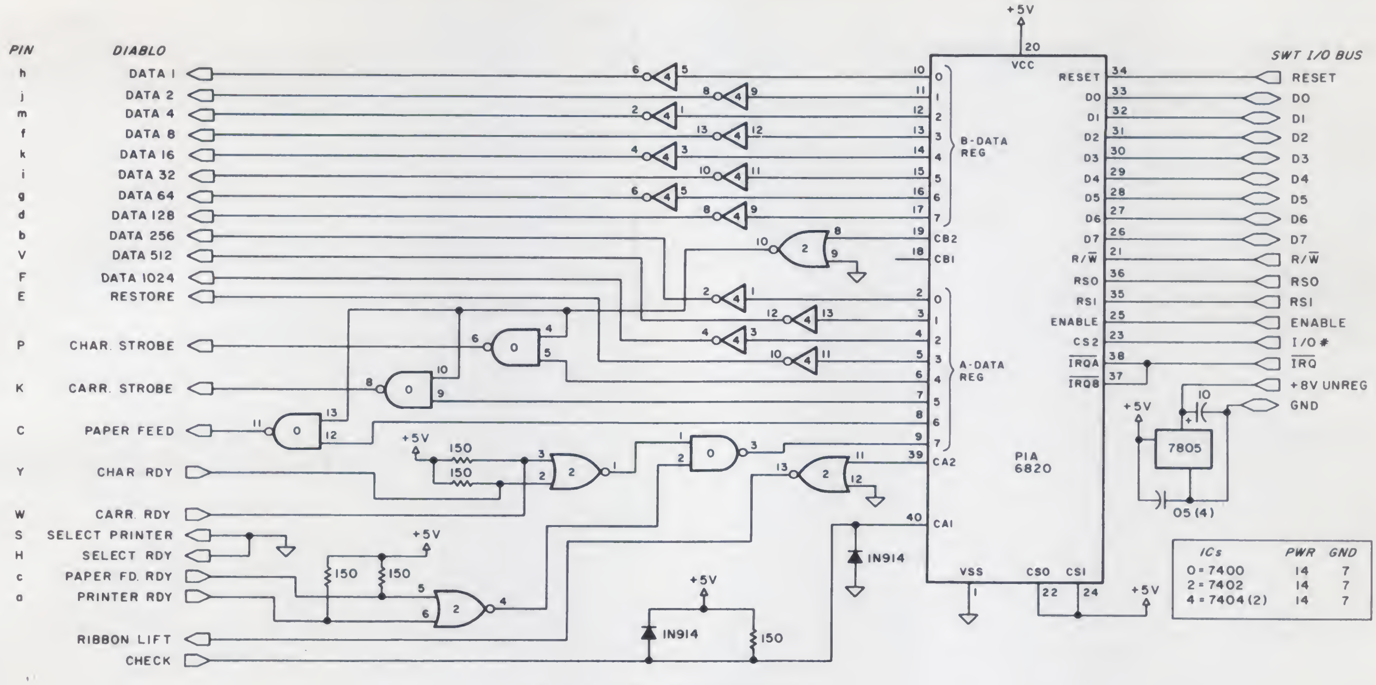


Fig. 1. SWTP/Diablo interface.

by setting the CA2 control line. A restore command is then sent to the printer; the character and line counters are reset; and control is passed back to the calling program.

The routine that actually performs the printing is HOUT, which looks like the SWTBUG routine OUTEEE to the caller but sends the desired character to the Diablo printer instead of the control terminal. On entry to HOUT there are three possible conditions for the passed character: a control character, a space or a printable character. After waiting for the printer to return a ready status, HOUT checks for these three possibilities.

If the character is a space (hexadecimal 20), the character count will be incremented; a carriage move command for one character width will be issued; and control will be returned to the calling program. If the character is printable (greater than hexadecimal 20), it will be strobed into the printer character register, and then the space routine will be entered to update the character count, move the carriage one space and then return control to the

```
B07A 26 01               BNE    HLF100
B07C 4F         HLF050   CLR A
B07D B7 B0 F6   HLF100   STA A  HLNCNT
B080 20 C3               BRA    HRET        RESTORE AND RETURN

B082 B6 B0 D6   HFF      LDA A  HLPPCT
B085 B0 B0 F6            SUB A  HLNCNT
B088 C6 08               LDA B  #LINMOV     INCS/LINE
B08A BD B0 AA            JSR    MULT
B08D 8A 40               ORA A  #$40
B08F A7 00               STA A  0,X
B091 E7 02               STA B  2,X
B093 20 E7               BRA    HLF050
B095            HRDY     EQU    *           WAIT FOR READY (AND CHECK FOR CHECK)
B095 CE 80 1C            LDX    #HPIA       GET PORT ADDRESS
B098 6D 01      HRWAIT   TST    1,X         Printer check?
B09A 2B 05               BMI    HCHK        Yes
B09C 6D 00               TST    0,X         CHECK FOR READY
B09E 2B F8               BMI    HRWAIT      WAIT FOR READY
B0A0 39                  RTS
B0A1            HCHK     EQU    *           PRINTER CHECK - ABORT
B0A1 CE B0 D7            LDX    #ABTMSG
B0A4 BD E0 7E            JSR    PDATA1      PRINT ERROR MESSAGE
B0A7 7E 71 03            JMP    WARMS       RETURN TO FLEX

                * REENTRANT MULTIPLICATION
                * A * B =A,B
B0AA 36         MULT     PSH A
B0AB 86 08               LDA A  #8
B0AD 36                  PSH A
B0AE 30                  TSX
B0AF 4F                  CLR A
B0B0 56                  ROR B
B0B1 24 02      M3       BCC    M4          DO NOT ADD
B0B3 AB 01               ADD A  1,X         ADD MULTIPLICAND
B0B5 46         M4       ROR A
B0B6 56                  ROR B
B0B7 6A 00               DEC    0,X         DEC COUNT
B0B9 26 F6               BNE    M3
B0BB 31                  INS
B0BC 31                  INS
B0BD CE 80 1C            LDX    #HPIA       RESTORE PIA ADDRESS
B0C0 39                  RTS

                * THIS IS THE ACTUAL TEST ROUTINE
B0C1            GO       EQU    *
B0C1 BD B0 03            JSR    HINIT
B0C4 BD E1 AC   BACK     JSR    INEEE       GET CHARACTER
B0C7 81 18               CMP A  #TERMC      TERMINATION CHARACTER?
B0C9 27 08               BEQ    DONE        YES
B0CB BD E1 D1            JSR    OUTEEE
B0CE BD B0 27            JSR    HOUT
B0D1 20 F1               BRA    BACK
B0D3 7E 71 03   DONE     JMP    WARMS
B0D6 42         HLPPCT   FCB    LPP         LINES PER PAGE
B0D7 2D         ABTMSG   FCC    /-->Printer Check -- ABORT
B0D8 2D 3E
B0DA 50 72
B0DC 69 6E
B0DE 74 65
B0E0 72 20
B0E2 43 68
B0E4 65 63
B0E6 6B 20
B0E8 2D 2D
B0EA 20 41
B0EC 42 4F
B0EE 52 54
B0F0 07                  FCB    7,7,4       BEL, BEL, EOT
B0F1 07 04
B0F3            HSAVEX   RMB    2           SAVE USER X
B0F5            HCHCNT   RMB    1           CURRENT CHARACTER COUNT
B0F6            HLNCNT   RMB    1           LINES ON THIS PAGE

 SYMBOL TABLE:

ABTMSG B0D7   BACK   B0C4   CHRMOV 0006   CRAMSK 003C   CRBMSK 002C
DONE   B0D3   GO     B0C1   HCHCNT B0F5   HCHK   B0A1   HCNTRL B04A
HCR    B058   HFF    B082   HINIT  B003   HLF    B06B   HLF050 B07C
HLF100 B07D   HLNCNT B0F6   HLPPCT B0D6   HOUT   B027   HPIA   801C
HRDY   B095   HRET   B045   HRWAIT B098   HSAVEX B0F3   HSPACE B03A
INEEE  E1AC   LINMOV 0008   LPP    0042   M3     B0B1   M4     B0B5
MULT   B0AA   OUTEEE E1D1   PDATA1 E07E   TERMC  0018   VERSIO 0003
WARMS  7103
```

calling program.

If the character is a control function (less than hexadecimal 20), it will be checked to determine if it is a carriage return, line feed or form feed. If it is not any of these three control characters, it will be ignored. If the character is a carriage return, then the character count will be multiplied by the character width and the product will be strobed into the carriage command register along with the direction bit. This causes the carriage to back-space the same number of spaces it has moved forward on this line. The character count is then reset, and control is returned to the calling program.

If the character is a line feed, then the line height count will be strobed into the paper feed register. The line count will be incremented and reset to zero if it is equal to the page length. Control will then be returned to the calling program. If the character is a form feed, the number of lines remaining on the page will be computed, multiplied by the line height and strobed into the paper feed register. The line count will then be reset, and control will be returned to the calling program.

HRDY is a utility routine called by HOUT. HRDY checks for a printer check and then waits for PRINTER READY before returning. If a check occurs, an error message will be printed on the control terminal and con-

Listing 2. HYDRIV—Diablo HyType driver.

```
                  * HYDRIV FOR TSC MINI-FLEX 6-27-79 @ 1600
                  * EQUs
0007              VERSION EQU   7          PROGRAM VERSION NUMBER
7118              PSTRNG  EQU   $7118      FLEX print string
7103              WARMS   EQU   $7103      FLEX warmstart
7124              RSTRIO  EQU   $7124      Restore I/O vectors
0006              CHRMOV  EQU   6          1/10 inch char spacing
0008              LINMOV  EQU   8          6 lines per inch
0042              LPP     EQU   66         lines per page
801C              HPIA    EQU   $801C      PORT 7
003C              CRAMSK  EQU   $00111100  CONTROL REG A MASK
002C              CRBMSK  EQU   $00101100  CONTROL REG B MASK

B000                      ORG   $B000
B000 20 01                BRA   HINIT      Humor VERSION
B002 07                   FCB   VERSION    PROGRAM VERSION NUMBER
B003              HINIT   EQU   *          PORT AND PRINTER INITIALIZATION
B003 CE 80 1C             LDX   #HPIA      GET PORT ADDRESS
                  * Setup data direction registers
B006 6F 01                CLR   1,X        SELECT DDRA
B008 6F 03                CLR   3,X        SELECT DDRB
B00A 86 7F                LDA A #$7F       1 IN, 7 OUT
B00C A7 00                STA A 0,X        Set DDRA
B00E 86 FF                LDA A #$FF       8 OUTPUTS
B010 A7 02                STA A 2,X        Set DDRB
B012 86 3C                LDA A #CRAMSK    SETUP CONTROL REGISTERS
B014 A7 01                STA A 1,X        Set CRA
B016 86 2C                LDA A #CRBMSK
B018 A7 03                STA A 3,X        Set CRB
B01A 86 08                LDA A #$08       RESTORE PRINTER
B01C A7 00                STA A 0,X
B01E 6F 00                CLR   0,X
B020 7F B0 E3             CLR   HCHCNT     Clear character
B023 7F B0 E4             CLR   HLNCNT     and line counters
B026 39                   RTS
B027              HOUT    EQU   *          OUTPUT CHARACTER ROUTINE
B027 FF B0 E1             STX   HSAVEX     SAVE USER X
B02A BD B0 95             JSR   HRDY       CHECK FOR READY
B02D 37                   PSH B
B02E 81 20                CMP A #$20       SPACE OR CONTROL?
B030 27 08                BEQ   HSPACE     SPACE
B032 2D 16                BLT   HCNTRL     CONTROL
B034 C6 10                LDA B #$10       SETUP CHAR STROBE FUNCTION
B036 E7 00                STA B 0,X
B038 A7 02                STA A 2,X        PRINT CHAR
B03A              HSPACE  EQU   *
B03A 7C B0 E3             INC   HCHCNT     INC CHAR COUNT
B03D C6 20                LDA B #$20       carrage move
B03F 86 06                LDA A #CHRMOV    1 character space
B041 E7 00                STA B 0,X
B043 A7 02                STA A 2,X
B045              HRET    EQU   *          RESTORE & RETURN
B045 33                   PUL B
B046 FE B0 E1             LDX   HSAVEX     RESTORE USER X
B049 39                   RTS
B04A              HCNTRL  EQU   *          HANDLE CONTROL CHARS
B04A 81 0D                CMP A #$D        CR?
B04C 27 0A                BEQ   HCR
B04E 81 0A                CMP A #$A        LF?
B050 27 19                BEQ   HLF
B052 81 0C                CMP A #$C        FF?
B054 27 2C                BEQ   HFF
B056 20 ED                BRA   HRET       forget it

B058 F6 B0 E3     HCR     LDA B HCHCNT     get char count
B05B 86 06                LDA A #CHRMOV    char length
B05D BD B0 AD             JSR   MULT       figure distance
B060 8A 24                ORA A #$24       carrage strobe & direction
B062 A7 00                STA A 0,X
B064 E7 02                STA B 2,X
B066 7F B0 E3             CLR   HCHCNT     clear char coun
B069 20 DA                BRA   HRET       return
B06B 86 40        HLF     LDA A #$40       paper feed strobe
B06D A7 00                STA A 0,X
B06F 86 08                LDA A #LINMOV    Increments/line
B071 A7 02                STA A 2,X
B073 B6 B0 E4             LDA A HLNCNT     GET LINE COUNT
B076 4C                   INC A            INCREMENT
B077 B1 B0 C4             CMP A HLPPCT
B07A 26 01                BNE   HLF100
B07C 4F           HLF050  CLR A
B07D B7 B0 E4     HLF100  STA A HLNCNT
B080 20 C3                BRA   HRET       RESTORE AND RETURN

B082 B6 B0 C4     HFF     LDA A HLPPCT
B085 B0 B0 E4             SUB A HLNCNT
B088 C6 08                LDA B #LINMOV    INCS/LINE
B08A BD B0 AD             JSR   MULT
B08D 8A 40                ORA A #$40
B08F A7 00                STA A 0,X
B091 E7 02                STA B 2,X
B093 20 E7                BRA   HLF050
B095              HRDY    EQU   *          WAIT FOR READY (AND CHECK FOR CHECK)
B095 CE 80 1C             LDX   #HPIA      GET PORT ADDRESS
B098 6D 01        HRWAIT  TST   1,X        Printer check?
B09A 2B 05                BMI   HCHK       Yes
B09C 6D 00                TST   0,X        CHECK FOR READY
B09E 2B F8                BMI   HRWAIT     WAIT FOR READY
B0A0 39                   RTS
B0A1              HCHK    EQU   *          PRINTER CHECK - ABORT
B0A1 BD 71 24             JSR   RSTRIO     Restore I/O vectors
B0A4 CE B0 C5             LDX   #ABTMSG
B0A7 BD 71 18             JSR   PSTRNG     PRINT ERROR MESSAGE
B0AA 7E 71 03             JMP   WARMS      RETURN TO FLEX

                  * REENTRANT MULTIPLICATION
                  * A * B =A,B
B0AD 36           MULT    PSH A
B0AE 86 08                LDA A #8
B0B0 36                   PSH A
B0B1 30                   TSX
B0B2 4F                   CLR A
B0B3 56                   ROR B
B0B4 24 02        M3      BCC   M4         DO NOT ADD
B0B6 AB 01                ADD A 1,X        ADD MULTIPLICAND
B0B8 46           M4      ROR A
B0B9 56                   ROR B
B0BA 6A 00                DEC   0,X        DEC COUNT
B0BC 26 F6                BNE   M3
B0BE 31                   INS
B0BF 31                   INS
B0C0 CE 80 1C             LDX   #HPIA      RESTORE PIA ADDRESS
B0C3 39                   RTS

B0C4 42           HLPPCT  FCB   LPP        LINES PER PAGE
B0C5 2D           ABTMSG  FCC   /-->Printer Check -- ABORT
B0C6 2D 3E
B0C8 50 72
B0CA 69 6E
B0CC 74 65
B0CE 72 20
B0D0 43 68
B0D2 65 63
B0D4 6B 20
B0D6 2D 2D
B0D8 20 41
B0DA 42 4F
B0DC 52 54
B0DE 07                   FCB   7,7,4      BEL, BEL, EOT
B0DF 07 04
B0E1              HSAVEX  RMB   2          SAVE USER X
B0E3              HCHCNT  RMB   1          CURRENT CHARACTER COUNT
B0E4              HLNCNT  RMB   1          LINES ON THIS PAGE
0010                      ORG   $10        Setup printer vectors in FLEX
0010 B0 03                FDB   HINIT      Initialization
0012 B0 27                FDB   HOUT       Put character
710D                      ORG   $710D
710D B0 27                FDB   HOUT       Put character
                          END

SYMBOL TABLE:

ABTMSG B0C5  CHRMOV 0006  CHAMSK 003C  CRBMSK 002C  HCHCNT B0E3
HCHK   B0A1  HCNTRL B04A  HCR    B058  HFF    B082  HINIT  B003
HLF    B06B  HLF050 B07C  HLF100 B07D  HLNCNT B0E4  HLPPCT B0C4
HOUT   B027  HPIA   801C  HRDY   B095  HRET   B045  HRWAIT B098
HSAVEX B0E1  HSPACE B03A  LINMOV 0008  LPP    0042  M3     B0B4
M4     B0B8  MULT   B0AD  PSTRNG 7118  RSTRIO 7124  VERSIO 0007
WARMS  7103
```

trol will be returned to the monitor.

HYTEST (Listing 1) echoes characters entered from the control terminal on the Diablo printer. It uses the routines HINIT and HOUT previously described. The actual test routine starts at label GO. All characters entered are echoed to the control terminal and sent to the printer. Entering a control X (the value of symbol TERMC) causes control to be returned to the monitor.

The Mini-FLEX interface consists of the HINIT and HOUT routines with additional code to set up the I/O vectors within the operating system. Listing 2 shows this version of the Diablo driver. If its binary is saved on the system disk as PRINT.SYS, then the P command will cause it to be loaded and used as the printer driver. Note that if a printer malfunction (CHECK) occurs, then routine HCHK will restore the FLEX I/O vectors before printing the error message and terminating.

The FLEX 2.0 interface was the most complicated because of the requirements to support print spooling. Three routines are required: the initialization

Listing 3. HYDRIV—Diablo driver for FLEX 2.0.

```
                    * HYDRIV FOR FLEX 2.0 6-27-79 @ 1900
                    * EQUs
AD1E                PSTRNG  EQU    $AD1E     FLEX print string
AD03                WARMS   EQU    $AD03     FLEX warmstart
E07E                PDATA1  EQU    $E07E     SWTBUG print string - Used for printer
0006                CHRMOV  EQU    6         1/10 inch char spacing
0008                LINMOV  EQU    8         6 lines per inch
0042                LPP     EQU    66        lines per page
801C                HPIA    EQU    $801C     PORT 7
003C                CRAMSK  EQU    $00111100 CONTROL REG A MASK
002C                CRBMSK  EQU    $00101100 CONTROL REG B MASK

ACC0                        ORG    $ACC0     PRINTER INITIALIZATION
ACC0 7E C0 00               JMP    HINIT     jump to it -- it won't fit
ACD8                        ORG    $ACD8     PRINTER READY CHECK
ACD8 36                     PSH A            save A
ACD9 B6 80 1D               LDA A  HPIA+1    get status
ACDC 43                     COM A            set CC > -=ready, +=not ready
ACDD 32                     PUL A
ACDE 39                     RTS
ACE4                        ORG    $ACE4     PRINT CHARACTER IN A
ACE4 7E C0 24               JMP    HOUT
C000                        ORG    $C000     PUT THE CODE THAT DOESN'T FIT HERE
C000                HINIT   EQU    *         PORT AND PRINTER INITIALIZATION
C000 CE 80 1C               LDX    #HPIA     GET PORT ADDRESS
C003 6F 01                  CLR    1,X       SELECT DDRA
C005 6F 03                  CLR    3,X       SELECT DDRB
C007 86 7F                  LDA A  #$7F      1 IN, 7 OUT
C009 A7 00                  STA A  0,X       Set DDRA
C00B 86 FF                  LDA A  #$FF      8 OUTPUTS
C00D A7 02                  STA A  2,X       Set DDRB
C00F 86 3C                  LDA A  #CRAMSK   SETUP CONTROL REGISTERS
C011 A7 01                  STA A  1,X       CRA
C013 86 2C                  LDA A  #CRBMSK
C015 A7 03                  STA A  3,X       CRB
C017 86 08                  LDA A  #$08      RESTORE PRINTER
C019 A7 00                  STA A  0,X
C01B 6F 00                  CLR    0,X
C01D 7F C0 C8               CLR    HCHCNT    Clear character
C020 7F C0 C9               CLR    HLNCNT    and line counters
C023 39                     RTS

C024                HOUT    EQU    *         OUTPUT CHARACTER ROUTINE
C024 FF C0 C6               STX    HSAVEX    SAVE USER X
C027 BD C0 92               JSR    HRDY      CHECK FOR READY
C02A 37                     PSH B
C02B 81 20                  CMP A  #$20      SPACE OR CONTROL?
C02D 27 08                  BEQ    HSPACE    SPACE
C02F 2D 16                  BLT    HCNTRL    CONTROL
C031 C6 10                  LDA B  #$10      SETUP CHAR STROBE FUNCTION
C033 E7 00                  STA B  0,X
C035 A7 02                  STA A  2,X       PRINT CHAR
C037                HSPACE  EQU    *
C037 7C C0 C8               INC    HCHCNT    INC CHAR COUNT
C03A C6 20                  LDA B  #$20      carrage move
C03C 86 06                  LDA A  #CHRMOV   1 character space
C03E E7 00                  STA B  0,X
C040 A7 02                  STA A  2,X
C042                HRET    EQU    *         RESTORE & RETURN
C042 33                     PUL B
C043 FE C0 C6               LDX    HSAVEX    RESTORE USER X
C046 39                     RTS

C047                HCNTRL  EQU    *         HANDLE CONTROL CHARS
C047 81 0D                  CMP A  #$D       CR?
C049 27 0A                  BEQ    HCR
C04B 81 0A                  CMP A  #$A       LF?
C04D 27 19                  BEQ    HLF
C04F 81 0C                  CMP A  #$C       FF?
C051 27 2C                  BEQ    HFF
C053 20 ED                  BRA    HRET      forget it

C055 F6 C0 C8       HCR     LDA B  HCHCNT    get char count
C058 86 06                  LDA A  #CHRMOV   char length
C05A BD C0 A7               JSR    MULT      figure distance
C05D 8A 24                  ORA A  #$24      carrage strobe & direction
C05F A7 00                  STA A  0,X
C061 E7 02                  STA B  2,X
C063 7F C0 C8               CLR    HCHCNT    clear char coun
C066 20 DA                  BRA    HRET      return

C068 86 40          HLF     LDA A  #$40      paper feed strobe
C06A A7 00                  STA A  0,X
C06C 86 08                  LDA A  #LINMOV   Increments/line
C06E A7 02                  STA A  2,X
C070 B6 C0 C9               LDA A  HLNCNT    GET LINE COUNT
C073 4C                     INC A            INCREMENT
C074 B1 C0 BE               CMP A  HLPPCT    Equal to lines/page?
C077 26 01                  BNE    HLF100    No
C079 4F             HLF050  CLR A            Yes -- Clear line count
C07A B7 C0 C9       HLF100  STA A  HLNCNT
C07D 20 C3                  BRA    HRET      RESTORE AND RETURN

C07F B6 C0 BE       HFF     LDA A  HLPPCT
C082 B0 C0 C9               SUB A  HLNCNT
C085 C6 08                  LDA B  #LINMOV   INCS/LINE
C087 BD C0 A7               JSR    MULT
C08A 8A 40                  ORA A  #$40
C08C A7 00                  STA A  0,X
C08E E7 02                  STA B  2,X
C090 20 E7                  BRA    HLF050

C092                HRDY    EQU    *         WAIT FOR READY (AND CHECK FOR CHECK)
C092 CE 80 1C               LDX    #HPIA     GET PORT ADDRESS
C095 6D 01          HRWAIT  TST    1,X       Printer check?
C097 2B 05                  BMI    HCHK      Yes
C099 6D 00                  TST    0,X       CHECK FOR READY
C09B 2B F8                  BMI    HRWAIT    WAIT FOR READY
C09D 39                     RTS
C09E                HCHK    EQU    *         PRINTER CHECK
C09E CE C0 BF               LDX    #BELLS    Get string address
C0A1 BD E0 7E               JSR    PDATA1    Send some bells
C0A4 7E AD 03               JMP    WARMS     Give up and abort everything

                    * REENTRANT MULTIPLICATION
                    * A * B =A,B
C0A7 36             MULT    PSH A
C0A8 86 08                  LDA A  #8
C0AA 36                     PSH A
C0AB 30                     TSX
C0AC 4F                     CLR A
C0AD 56                     ROR B
C0AE 24 02          M3      BCC    M4        DO NOT ADD
C0B0 AB 01                  ADD A  1,X       ADD MULTIPLICAND
C0B2 46             M4      ROR A
C0B3 56                     ROR B
C0B4 6A 00                  DEC    0,X       DEC COUNT
C0B6 26 F6                  BNE    M3
C0B8 31                     INS
C0B9 31                     INS
C0BA CE 80 1C               LDX    #HPIA     RESTORE PIA ADDRESS
C0BD 39                     RTS

C0BE 42             HLPPCT  FCB    LPP       LINES PER PAGE
C0BF 07             BELLS   FCB    7,7,7,7,7,7,4 SOME BELLS
C0C0 07 07
C0C2 07 07
C0C4 07 04
C0C6                HSAVEX  RMB    2         SAVE USER X
C0C8                HCHCNT  RMB    1         CURRENT CHARACTER COUNT
C0C9                HLNCNT  RMB    1         LINES ON THIS PAGE
                            END

SYMBOL TABLE:

BELLS  C0BF  CHRMOV 0006  CRAMSK 003C  CRBMSK 002C  HCHCNT C0C8
HCHK   C09E  HCNTRL C047  HCR    C055  HFF    C07F  HINIT  C000
HLF    C068  HLF050 C079  HLF100 C07A  HLNCNT C0C9  HLPPCT C0BE
HOUT   C024  HPIA   801C  HRDY   C092  HRET   C042  HRWAIT C095
HSAVEX C0C6  HSPACE C037  LINMOV 0008  LPP    0042  M3     C0AE
M4     C0B2  MULT   C0A7  PDATA1 E07E  PSTRNG AD1E  WARMS  AD03
```

An Introduction to Small Business Software for the PET

An Overview of an Inventory and Mailing List Maintenance System.

It's about time! You've been seeing ads for small business software in just about every computer magazine published. With prices in the range of $25 to thousands of dollars how does one decide just which system is the best to use. At best, a difficult choice—at worst, can bring on sleepless nights.

Now after long hours of thorough research and many months of programming, DR. DALEY's Software has come up with the first installment of a complete small business software system. However, it has some major disadvantages that you should consider before spending any of your hard earned cash—but more on that later.

DESIGN PHILOSOPHY

The first program is an inventory maintenance system. This is followed by a mailing list program. One of the first things learned in the research on an inventory program is that, despite the textbooks, virtually every small business operation has different requirements for its inventory information. This, of course, means that every business would either have to modify the packaged programs that it purchased, or hire its own consultant to write a custom program. It seems to us that either approach is unsatisfactory. The first would require considerable time and expertise, while the latter would be very expensive.

Another option is to write prepackaged software which each individual user can configure to his own needs. This would allow each business to customize its own computer maintained inventory files to, as closely as is possible, parallel the current inventory operation.

The approach selected for the design of the inventory system was to write a program which would allow the user to design, within reasonable limits, the configuration of the computer files and all operations on these files. This means that the user can computerize the business operations with less of the anguish that frequently accompanies this conversion.

After the design approach is selected, the task of coding the program is begun. The main thought in the coding process is to make the operation as easy and flexible as possible. Give the user the greatest conceivable number of useful operations and support these with various hard copy reports. Finally, be sure that the capacity of the system is sufficient to allow most any business to make use of it.

In summary, the operations of the inventory system will allow the user flexibility to design and maintain useful files which look like the files he already uses in his business. It will also allow reasonably large capacity with each of the 2010 records on a diskette having a total of 79 USABLE characters.

IMPLEMENTATION

The total operation of the system is "menu" driven with a number of "plain English" menu options. These options include adding records, editing them and saving them to the files. Also one can see, or edit individual records once they are placed on the disk. In addition one can zero a particular field on the disk for all records and calculate the value of the inventory for the entire inventory or for virtually any conceivable subset of the file. Finally one can obtain a listing of the entire file or almost any possible subset. For convenience a disk maintenance program is included which will allow you to copy files and to validate the integrity of the disk surface.

The one feature which sets this inventory system apart is the "Group search function" option. This option will allow the user to search through the files for virtually any set of the files that he might wish to find. The operation will allow the user to specify up to three fields within each record to be used for the search keys. Each search key uses a pattern matching search. That is, one must have an exact match for locations specified in the search key. However, the pattern must also match. Thus one can search through the file for a specific pattern within each of up to three fields for the record. One can specify patterns as follows:

**P*9Z
this matches with $0P-9Z
and #/P29Z
and 16P:9Z

Thus one can select virtually any subset of the files by the appropriate selection of the search keys.

This does not really cover the entire operations on the files, but space simply does not allow the complete description of the system.

DISADVANTAGES

We warned you about this. This could easily discourage all but the most determined of you. Please consider these carefully before purchasing this product. Here they are:

1. You will have to do your own work in setting up the files. The programmer has not done this thinking for you. If you do not spend some time thinking about this, you will find that some of the operations described above will not really be of much use to you.

2. The system is only available in the Commodore model 2040 disk format. If you don't own this powerful computer, then you won't be able to use this inventory system. If you have some other brand of computer please turn the page, otherwise read on.

3. The printer output is designed around the features of the CBM model 2022 printer. If you choose to use another printer, then you are on your own in modifying the printer output routines. The programmer made this somewhat easier in that the printer routines are all written as subroutines, thus changes in one location can cover most of the modifications necessary.

4. You probably will have to purchase this program by mail directly from the author. Most computer stores have not, as yet, responded to our calls for dealers.

5. At the present time this program is not interactive with any computer accounting system. This will make the cost control with the inventory only somewhat easier than doing this by hand. This should be remedied by midsummer of 1980.

ORDERING

Those of you who will accept these disadvantages and work around them will want to order a copy for your business. This can be done by either persuading your dealer to order it for you or calling us directly at the number given below. The price is $99.95 plus 4 percent tax in the state of Michigan.

MAILING LIST

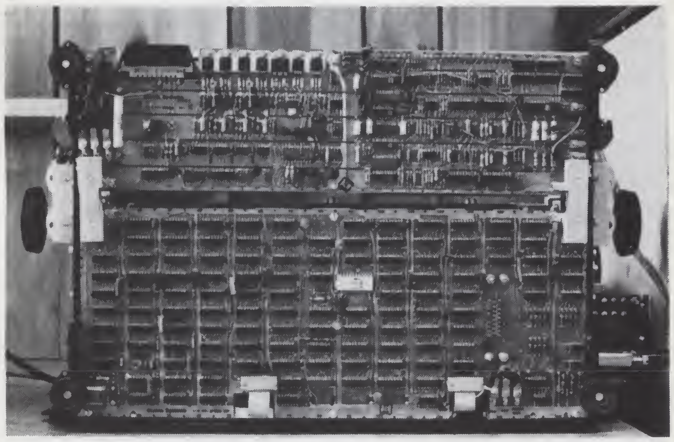
No we didn't forget this, but ran out of room. However, this program is much like the inventory system. One can have a total of 1340 names on a diskette with multiple diskettes in a file. The files are kept in sequence using any of the fields as a sort key. There is a practical limit with a 32K PET of about 125 diskettes. The user can design the appearance of the printer output. Almost any subset of the file can be printed. The price here is $99.95.

Charge to your MC/VISA

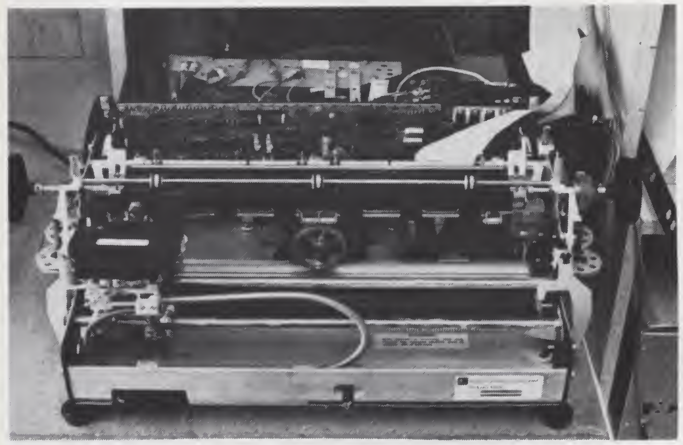
master charge THE INTERBANK CARD

VISA

DR. DALEY's Software 34
425 Grove Ave., Berrien Springs, MI 49103
Phone (616) 471-5514
Sun.-Thurs. noon to 9 p.m., Eastern Time



Diablo printer (bottom view).



Diablo printer (top view).

routine, the character output routine and a status routine, which returns a flag to indicate if the printer is ready.

Space is available in FLEX 2.0 for the routines, but it is not large enough for all the code necessary. Listing 3 shows the final result. All code that didn't fit in the reserved area was located starting at address C000. JMP instructions were placed in the areas reserved for the initialization and output routines to transfer control of the actual routines.

Note that if the printer drops READY, the check routine will just keep returning a not READY status. If the printer is running in the print spooling mode, the main task will continue to run. If the printer gets a CHECK condition after the ready routine has returned a READY status, then six bell characters will be sent to the control terminal and everything will be aborted by a return to the FLEX warm-start address.

Conclusion

So far, this is all that I have implemented, but I plan to include bidirectional printing, graphics and justification. Bidirectional printing saves the carriage return time, thus making the effective print speed greater.

In order to implement bidirectional printing, you must have a buffer big enough to save a complete line. After printing one line from left to right, you save the next line in the buffer. When you have all of that line, you figure its length, position the carriage and start printing it backward.

Once bidirectional printing is accomplished, adding justification should be easy. It could be implemented by either inserting extra spaces between words before printing the line or by adding spaces between the letters. This is possible because the carriage can be moved in 1/60 inch increments.

Graphics is a half-baked idea. The print mechanism is capable of spacing 1/60 inch in the horizontal direction and 1/48 inch in the vertical direction. Until I decide what I want the graphics to do, it will remain a fun idea in my head. ■

References

"Model 1200 HyType I Printer Product Description," Diablo Systems Incorporated, Hayward CA, 1975.

"Model 1200 HyType I Printer Maintenance Manual," Diablo Systems, Hayward CA, 1974.

"M6800 Systems Reference and Data Sheets," Motorola Semiconductor Products, Phoenix AZ, 1975.

"Mini-FLEX Ver. 1.0 Advanced Programmer Guide," Southwest Technical Products Corporation, San Antonio TX, 1978.

"SWTBUG 6800 ROM Monitor Version 1.0 Users Guide," Southwest Technical Products Corporation, San Antonio TX, 1977.

"FLEX Programmer's Manual," Technical Systems Consultants, Inc., West Lafayette IN, 1978.

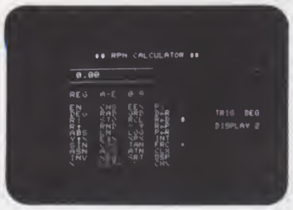
Software for the PET

PROGRAMMA Software Program Products

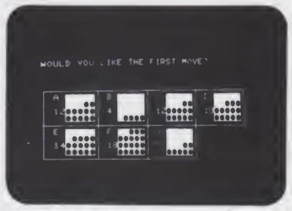
DOMINOES $ 6.95

OTHELLO $ 9.95

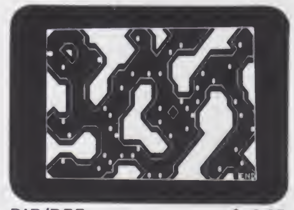
LETTER SQUARES $ 6.95



RPN MATHPACK $19.95



SUPER NIM $ 6.95



DIR/REF $ 6.95

SPACE WARS $ 9.95

GRAND PRIX $ 6.95

E.S.P. $ 9.95

FORECAST $ 9.95

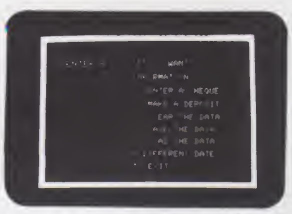
$ 6.95

BLOCKADE $ 9.95

DEPTH CHARGE $ 9.95

BASKETBALL $ 9.95

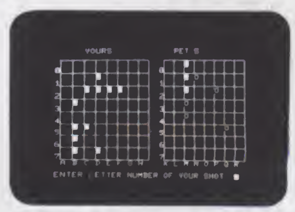
SLOT MACHINE $ 6.95



HOME ACCOUNTING $ 9.95

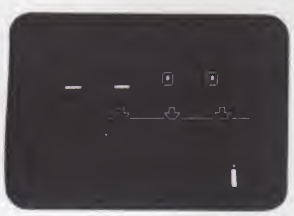
All orders include 3% postage and handling with a minimum of $1.00. California residents include 6% Sales Tax.

VISA MASTERCHARGE



BATTLE SHIP $ 9.95

PET IS A TRADEMARK OF COMMODORE BUSINESS MACHINES, INC.

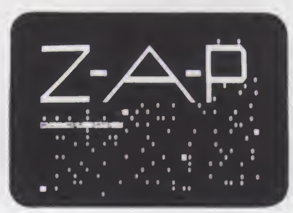


SHOOTING GALLERY $ 9.95

PROGRAMMA INTERNATIONAL, Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579
384-1116
384-1117

Dealer Inquiries Invited

✓11



ZAP $ 6.95

Dial-up Directory

In this installment, our intrepid author interviews the maintainers of the Chicago CBBS.

Frank J. Derfler, Jr.
PO Box 17283
Montgomery AL 36117

The people who bring you Computer Bulletin Board Services are a diverse lot. Their motivations range from purely mercenary to ultimately humanitarian. Some feel a strong responsibility for the material that is disseminated over their systems. Others believe in a "free press" and allow an uncensored flow of data to pass through their disks. The CBBS concept seems to be a spin-off of the commercial computer mail schemes, but it is much different in implementation. This month we will talk about these ideas and others with the two men who can truly be called the fathers of CBBS, Ward Christensen and Randy Suess.

I talked to Ward and Randy during a trip to Chicago. I went to Randy's home, which also houses the Chicago CBBS. We edged our way into Randy's basement, and while the system's disks clicked and whirled, Ward's disembodied voice joined us over a speakerphone from his home south of the city.

Randy Suess, the serious craftsman who is the hardware guru behind the Chicago CBBS. His software swami (Ward Christensen) was present as only a voice during the interview.

Microcomputing: Your own article in the November 1978 *Byte* gave an excellent technical description of the Chicago-style CBBS. What about the personal side? What were your goals and motivations in establishing the first of what has grown to be a series of systems?

Ward: My motivation arrived on the morning of January 16, 1978, coincidentally with the great Chicago snowstorm. I got up to find that my alley was impassable and started to think about more sedentary things.

Randy: We had previously both developed remote terminal operation for our systems so that we could use them when we were away from home. We started leaving messages for each other, and the idea grew.

Ward: Also, we had a regular physical bulletin board for our computer club, and computerizing it was an obvious move.

Microcomputing: Were you motivated by the mailbox services available on the ARPA net, PLATO or other commercial systems?

Ward: Honestly, I didn't know they existed at the time. I understand that we have reinvented the wheel by using some similar control codes, but I didn't know about them then. We started out with five functions. We made new version changes almost

weekly in the beginning, but the system has stayed pretty simple.

Randy: We used our own equipment, too. Eventually though, we had to dedicate a system so that we could provide full-time service. Now several manufacturers have donated equipment for use and evaluation. We have used every one of the S-100 modem boards available.

Microcomputing: Any comments about modem boards?

Randy: All of the manufacturers have been great. D.C. Hayes has been responsive to comments and recently helped the Dallas CBBS out with a problem. We are now running the Potomac Micro-Magic and are very happy with it.

Microcomputing: Is your user population still growing?

Ward: We started out using a Teletype for logging, and Randy used to send me hundreds of feet of paper at a time. Now we log on a separate disk. We have had over 11,000 users and are getting ten to 15 new folks calling in a day.

Microcomputing: Well, I can testify that you have the busiest phone number of any system.

Randy: The average caller stays on about 20 minutes, but expert users can get in and out in about five minutes. Our peak traffic loads are from 9 PM until early morning. We placed the system in a central Chicago location to cut down on the toll costs for our users, but it doesn't seem to matter. We get calls from across the country.

Microcomputing: What is your longest-distance user?

Randy: You have called from Hawaii, Frank, but we have had people log in from Australia. We have some European users, too.

Microcomputing: Well, a call to a busy CBBS is a quick way for people out of the country to get a feel for the latest microcomputing news and developments. With all of those diverse users, do you often have to play the role of policeman and censor the material?

Ward: Surprisingly, not often. It is easy to do, but we don't delete things very often. Trash on a system is self-perpetuating. If you catch it early, it doesn't grow. As you may have noted if you read the system sign-on, we try to keep the notices to computer-related subjects, so in that way we can exercise some discretion.

Randy: Cars for sale and computer dating don't really meet our definition of computer-related.

Microcomputing: How about some systems such as Boston, which has game players, or Beaverton, which has movie reviews?

Randy: That's great for them. They should get into chess or cars or anything else they want. We do understand that some people have gotten pretty good computer-related jobs through our system, and we are happy about that. We just want to keep the Chicago CBBS computer-related, so if we have to pack a disk, then car ads are the first to go.

Ward: We feel we have a responsibility as the first and probably the busiest system operating.

Microcomputing: Were you really the first?

Ward: The Kansas City Electronic Message System may have started at about the same time—I'm not really sure who got on first—but we continued to function.

Microcomputing: What other CBBSs around the country now use your software?

Ward: Boston, Atlanta, Dallas, Pasadena and Beaverton are operating. We have sold other copies, too.

Microcomputing: Are you really in the sales business?

Ward: Absolutely not! We had thought about giving the software away free, and then we thought about selling it for $25. But either way, we were afraid people would not value it and we would have no control over our creation. We settled on $50 as a fair price.

Randy: We have thousands and thousands of our own dollars in it, and it would take a lot of 50-dollar checks to turn a profit.

Microcomputing: What would it cost to start a CBBS right now?

Randy: You could easily do it for $2000. You do need a lot of disk space though. The Kansas City TRS-80 forum (816-861-7040) has some information on using a TRS-80 as a CBBS, I think, but their system is not derived from ours.

Ward: TRS-80 users have trouble with our system because they don't have the control codes that make the use of our system so easy.

Randy: TRS-80 users keep asking why we don't change our system for them. We recommend they ask the

---

The following list provides the location, phone number and other information about bulletin systems around the country. All have program exchange capability. I have personally checked into all of these systems; that is my only guarantee. I verified them from my list of over 110 "reported" systems.

LOCATION	PHONE NO.	COMMENTS
California		
Signal Hill	213-424-3506	6 PM–9 AM and weekends. ABBS software for sale.
Massachusetts		
Wellesley	617-431-1699	Not 24 hr. Forum-80.
Michigan		
Farmington Hills	313-477-4471	Not 24 hr. ABBS.
Southfield	313-569-2063	Detroit Apple Club.
Minnesota		
Minneapolis	612-929-8966	ABBS.
Texas		
Dallas	214-288-4859	FORUM-80.

Individual Listing

The following individuals have indicated a desire to exchange data calls for the purposes of chatting or swapping programs. Please call only during appropriate hours.

Leonard Garcia (214-522-1006) is the author of Telestar, a North Star terminal program. He has an extensive communications capability and will take data calls on Tuesdays and Thursdays and weekends from 7 PM to 10:30 PM Central time.

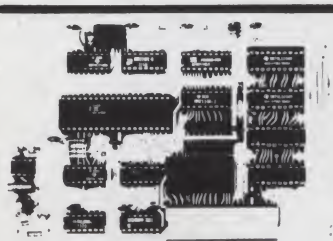
Tim Lovatt (206-482-5134) is interested in Apple program swaps.

Bill Crawford (615-877-7603) uses a TRS-80 with the ST-80D software. He is interested in computer clubs and ham radio. Call 6–9 PM Eastern time.

Jim Craft (703-386-3503) has a TRS-80 and is interested in ham radio software.

Chuck Dedman (216-282-4248) is interested in starting a bulletin board system in Ohio.

Donald Warren (404-834-4001) is available to chat after 6 PM Eastern time.



The busiest CBBS in the country. Notice the extensive shielding and air conditioning needed for such a high-technology installation.

manufacturer why standard ASCII control codes were not included in their product.

Ward: They always want us to fix the bars they get on the screen. The bars represent parity errors and their manual tells them how to get rid of them, but they still keep asking.

Microcomputing: Are we entering an era such as we saw in amateur radio when the home-brew tinkerers resented the operators of ready-made "appliances"?

Randy: Sure, but we certainly do get tired of answering the same non-problems.

Microcomputing: How about the future? Telenet has announced some super-low night rates for data transmission. Do you see any linking of systems for transfer of general-interest messages?

Randy: That would get us into long-distance calling. We deliberately use a dial-in-only line that costs three dollars a month. Message transfers would require a lot of time and software.

Ward: We intend that this system remain free of cost to the user. Nationwide netting might become complicated and expensive. Also, we find that individuals already transfer interesting information from system to system, or they may at least leave references to messages of interest on other systems. The same thing is true of becoming multi-user. We are frequently asked why we don't provide more lines and go multi-user. The best answer we can give is that we are only in this for the fun of it. Big changes will come slowly.

Microcomputing: What little things are you looking at?

Ward: Oh, a lot of housekeeping things. We need to keep our message numbers straight even after we pack a disk. Message number 110 should always remain number 110 and not suddenly become 29. Also, supporting 110 baud may be a time-wasting service. We also are considering a function that would allow swapping complete programs.

Microcomputing: Wouldn't program swapping be more easily done by direct person-to-person data calls, perhaps arranged on the CBBS?

Ward: Sure, and it would be easier on our disk space too.

Microcomputing: Any final comments?

Randy: This is just our hobby, but we do feel some responsibility to our users. We will keep on trying to enhance the system and respond to needs for new functions.

Ward: Amen, and tell your readers to conserve disk space.

Comments?

If you are a bulletin-service owner or user and have comments or items to discuss, let me know. Also let me know if you are interested in receiving direct data calls and briefly describe your interests, equipment capabilities and available times. We will make you part of the Dial-up Directory. Either drop me a line (PO Box 17283, Montgomery AL 36117) or leave a message for me on the Atlanta system (404-939-1520). ■

DOC

A Multipurpose Utility Package

Your North Star system gets a dose of the right medicine when DOC comes to call.

Charles Alliston
K-C Word Processing
263 Blueberry Lane
West Lafayette IN 47906

DOC (Documentation, Optimization and Confidentiality), a utility software system for North Star users from Mini Business Systems, PO Box 15587, Salt Lake City UT 84115, may have just the prescription for that ailing program in need of some attention to improve its performance. The system is a multipurpose programming aid providing documentation, compaction, improved efficiency and confidentiality in an easy-to-use, self-prompting sequence.

The value of each capability is proportional to the complexity and length of the program. For example, the cross-reference listing can be of special assistance when a program has become difficult to follow through several modifications, or if the user is attempting to analyze a program with which he is not familiar.

A run will provide an alphabetized listing of all variables and the line(s) in which each is referenced. This can, for example, be particularly useful when adding code, and the programmer needs to use a variable not already in use. In addition, all GOTO or GOSUB statements are similarly documented.

Implementation

The software is provided on a minifloppy and consists of three single-density programs in BASIC that have themselves been processed by DOC to provide confidentiality and optimization. In addition, there are three data files that contain the list of statements and variables used in the source program, as well as a list of arguments that reflect the options selected by the user.

It is necessary that both the program and data files be on drive one of the disk system. The "source" program may be located on another drive, as can the "target" file, which will contain the optimized program upon completion of the run. However, both must be set to "type 3" data files prior to running the program.

The documentation consists of 20 pages of suggestions and easily followed instructions for implementation. I encountered no difficulties using a North Star Horizon II and Release 5 Double-Density BASIC. Only small programs can be processed with 24K of memory because a total of only 170 combinations of references to variables, GOTO or GOSUB statements can be documented. If 32K is available (not counting memory below DOS), the capacity will increase to 700 and become 1200 with 40K of memory.

Operation

The Program listing shows a sample run of DOC using a trivial program with the printer as the input and output device. User responses are underlined. Only one program (BTPDO) need be loaded and run because it chains to the related programs. Following the sign-on message, the user responds to prompts and selects the desired options.

Documentation of a program is assisted by providing (1) a program listing on the chosen device with a user-supplied heading (see Program listing) and (2) cross-reference tables containing ordered listings of variables and GOTO- and GOSUB-type statements combined in one listing. This capability alone is a great asset for modifying current programs or

Program listing.

(Start "DOC" Session)

```
BYE
+TY SOURCE 3
+TY TARGET 3
+GO BASIC
READY
LOAD BTPDO
READY
RUN

***************************************************************
* D O C  -  DOCUMENTATION, OPTIMIZATION, AND CONCATENATION    *
```

understanding unfamiliar ones. This portion of the package is comparable to CBASIC's Cross-Reference Lister XREF or RSTS' CREF.

Programs may be optimized by concatenation of multiple statements into one line, deletion of REMARK statements and unnecessary blanks at the user's option. If a REMARK statement is referenced (see original line 40), it is not deleted (line 40, optimized version). The length of the optimized line is user controlled (maximum = 255 characters).

Although long line length may not be correctly listed on your output device, the program will operate, and efficiency in execution of GOTO-type statements is gained by reducing the total number of lines in the program. The sample program was optimized to 90 characters to facilitate listing in this article.

Provisions are made for avoiding concatenation when desired. The reduction in size will vary with programming methods. Since REMARKS were extensively used in the sample program, the reduction in size is not typical of most programs. If you optimize MAILER from North Star Software Exchange Six, for example, the reduction will be from 6250 bytes to 4381 bytes. Such compaction is, of course, a significant space saver.

In addition, it contributes to the fourth feature of DOC—confidentiality. Deletion of blanks makes pirating more difficult. However, a greater degree of protection is provided by the fact that when lines greater than 132 characters have been generated, the LIST and EDIT commands do not work properly, thus making a listing much more difficult to obtain.

With a Horizon II/ Soroc 120 / Diablo 1620 system, 99 seconds were required (from final RETURN to completion) for a run of the sample program, and 731 seconds for the MAILER program. Table 1 summarizes times required for the various phases of DOC for both programs.

While variables and GOTO-type statements are being

```
* COPYRIGHT 1978 BY MINI BUSINESS SYSTEMS                      *
* THIS PROGRAM MAY NOT BE COPIED IN ANY FORM WITHOUT PRIOR     *
* APPROVAL OF MINI BUSINESS SYSTEMS                            *
****************************************************************

THIS PROGRAM LISTS AND MAPS A BASIC PROGRAM FROM DISK
IF YOU DESIRE, IT ALSO COMPACTS & OPTIMIZES YOUR PROGRAM
IF YOU OPTIMIZE YOUR PROGRAM, THEN LISTING & EDITING IT MAY
BE IMPOSSIBLE IF YOU SELECT A LINE LENGTH GREATER THAN YOUR
TERMINAL SUPPORTS
THE PROGRAM FILE MUST BE TEMPORARILY SET TO TYPE 3
HOW MANY GOTO'S DO YOU ESTIMATE YOUR PROGRAM HAS ? 20
HOW MANY SEPARATE OCCURENCES OF VARIABLES DO YOU ESTIMATE ? 60
DO YOU WISH A CROSS REFERENCE LISTING ? YES


SOURCE FILE NAME (MUST TEMPORARILY BE SET TO TYPE 3) ?  SOURCE
LISTING TITLE? Demonstration Program
DATE? June 25, 1979
FORM LENGTH (LINES PER PAGE) 25
OUTPUT DEVICE # IS ? 0
ENTER 1 FOR SINGLE SPACE, 2 FOR DOUBLE, ETC. ? 1
DO YOU WISH TO OPTIMIZE THIS PROGRAM ? Y
WHAT IS THE MAXIMUM LINE LENGTH YOU DESIRE (MAX=255) ? 90
DO YOU WISH TO DELETE REM STATEMENTS ? Y
TARGET FILE NAME (TEMPORARILY TYPE 3)? TARGET
PUT PAPER AT HEAD OF FORM, AND PRESS RETURN TO CONTINUE

Demonstration Program              June 25, 1979            PAGE  1

 10 REM ***************************   DOCDEMO   ********************************
 20 REM
 30 GOTO  70 \                                      REM GO AROUND SUBROUTINE
 40 REM                                            INCLUDE A REFERENCED REMARK LINE
 50 L = LOG(N) \                                   REM USE LOG FUNCTION
 60 RETURN \                                       REM END OF SUBROUTINE
 70 INPUT "Your Name, Please? ", N$ \              REM GET USER'S NAME
 80 PRINT "Thank You, ", N$ \                      REM RESPOND TO USER
 90 INPUT "Please Give Me a Number: ",N \          REM OBTAIN A NUMBER FROM USER
 100 PRINT \ PRINT "OK, ", N$, \                   REM BEGIN OUTPUT
 110 PRINT " The Square Root Of ", N, " Is: ", SQRT(N) \ REM PRINT SQUARE ROOT
 120 GOSUB  40 \                                    REM USE A GOSUB STATEMENT
 130 PRINT "And the LOG is: ", L \                 REM PRINT LOG OF THE NUMBER
 140 PRINT \ INPUT"Use Another Number? ", Q$ \ REM ASK USER TO CONTINUE
 150 PRINT \ IF Q$ = "Y" THEN GOTO  90 \            REM USE A GOTO STATEMENT
 160 END


PLEASE WAIT - SORTING VARIABLES

Demonstration Program              June 25, 1979                    VARIABLE MAP

VARIABLE  USED IN LINE


L           50         130
N           50         90         110        110
N$          70         80         100
Q$          140        150


PLEASE WAIT - SORTING GOTO'S

Demonstration Program              June 25, 1979                    GOTO MAP

GOTO       FROM


 40        120
 70        30
 90        150


PLEASE WAIT - OPTIMIZATION PHASE
INPUT FILE WAS  892 BYTES LONG.
OUTPUT FILE IS  306 BYTES LONG.
OPTIMIZED PROGRAM IS IN FILE TARGET
          (End "Doc" Session: Optimized Program Listed with North Star "LIST" Command)
READY
BYE
+TY TARGET 2
+GO BASIC
READY
LOAD TARGET
READY
LIST

10GOTO70
40REM                                              INCLUDE A REFERENCED REMARK LINE
50L=LOG(N)\RETURN
70INPUT"Your Name, Please? ",N$\PRINT"Thank You, ",N$
90INPUT"Please Give Me a Number: ",N\PRINT\PRINT"OK, ",N$,
110PRINT" The Square Root Of ",N," Is: ",SQRT(N)
120GOSUB40\PRINT"And the LOG is: ",L\PRINT\INPUT"Use Another Number? ",Q$
150PRINT\IFQ$="Y"THENGOTO90\END
READY
```

	Program MAILER*	Name DOCDEMO
Original length (bytes)	6250	892
Optimized length (bytes)	4381	306
Original no. lines	244	16
Optimized no. lines	48	8
No. variables	33	4
No. references to variables	339	11
No. GOTO or GOSUB	37	3
No. references to GOTO, GOSUB	76	3
Time (sec.) required to:		
List program**	262	33
Sort variables	140	19
Print variables map**	61	4
Sort GOTO, GOSUB	28	5
Print GOTO, GOSUB map**	21	3
Optimize program	219	35
TOTAL TIME (seconds)	731	99

* From North Star Software Exchange 6.
**Using Diablo 1620 operating at 1200 baud setting.

Table 1. Summary of selected parameters for two programs.

sorted, a "PLEASE WAIT" (see Program listing) message is displayed on the monitor during intervals when the printer is inactive. The disk drive(s) also operates periodically to further reassure the operator that all is well.

You can accommodate future versions of North Star BASIC by including a feature to easily incorporate any unknown BASIC instruction into DOC's repertoire. The distribution disk is fully compatible with North Star Release 4 BASIC.

Also included with DOC is a well-documented program called GOTOSUB, which provides a more flexible GOTO N statement than ON N GO TO X1, X2, X3. Its use, along with additional hints on programming for maximum efficiency, should result in more efficient programs for the average user.

Finally, there is a "freebie" in the form of a Least Squares Curve Fitting Program intended as the source program for a sample run. Mini Business Systems included this program from the North Star User's Group Library.

It is surprisingly difficult to itemize undesirable features of DOC. In the interest of objectivity, I could be picky and comment on the inconvenience of having to set the source and target programs as data files (but then, that does provide a measure of protection!). It would be nice if the GOTO and GOSUB statements were separated. The choice of BTPDO as the name of the program to be loaded is not a very logical name, and somehow seems difficult for this absent-minded user to remember between sessions. (Recent information from MBS indicates that BTPDO stands for Basic Text Program Development and Optimization. DOC II revises the name and changes each of the items mentioned above as undesirable features. The current price of DOC is $60, a change from $29 when it was originally issued.)

In summary, the flexibility and multiple capabilities should make DOC a worthwhile investment for the North Star user who modifies existing software or develops new software. ■

✓Reader Service—see page 257

SCREEN

Two alternatives: modify an existing data-base management system to store and retrieve data, or write one for your own system. The author chose the latter; here are the results.

Forest E. Myers
5114 Garnett St.
Shawnee KS 66203

How many times have you sat down at the keyboard of your microcomputer to enter a program to store and retrieve data? I've done it numerous times to meet a wide variety of needs. In most cases, I structured programs in the same way: each program allowed the creation of a data file, correction, addition and deletion of records to a previously created data file and listing of data in the file.

I've repeated this process for our household accounting system, coupon files for my wife's myriad product coupons, for mailing lists, for my brother-in-law's realtor-reminder report, for my class attendance and test score files. I've become adept at cranking out these small data entry, update, storage and retrieval programs. However, with each one I did, the more I dreaded the prospects of doing another. Finally, I came to the conclusion that there must be a better way. Of course, there is a better way. There are numerous data base management packages available for microcomputers to take care of my needs. Unfortunately for me, no one ever bothered to write one for my particular system.

As a result, I had two options: I could buy, beg, borrow or steal a data base management system written for some other brand of microcomputer and modify it to run on my system, or I could sit down and write a small data base management program for my system.

After some deliberations, I decided on the latter alternative. Several factors prompted my decision. First, it was less expensive (basically my time). Second, it seemed that any software taken from another system might require considerable modification to run on my system.

This article outlines the system requirements to implement two programs that make up DATBAS, a data base management system. Additionally, it presents the first of the two DATBAS programs—SCREEN. The remaining program, FILEIT, is the subject of a follow-up article in next month's issue.

System Requirements

In order to set the stage for the discussion to follow and to help you understand the reason for some of the coding used in SCREEN, I'll briefly describe the computer system and the BASIC interpreter under which it was written. It should be emphasized at the outset that your system does not have to have all the features mentioned to implement DATBAS.

DATBAS was written on a microcomputer with 50K RAM, CRT, standard 8 inch floppy-disk drives and TTY-43. The 50K RAM, after allowance for the disk operating system and BASIC interpreter, translates into approximately 28K of user memory. If your system has less available user memory, it simply means that you must cut back on the size of arrays used.

The CRT displays 64 characters per line and 16 lines per CRT screen. If your display is 80 × 24, then you can get more information displayed on a given data entry screen. The floppy-disk storage device is not necessary. Only the ability to store data on some mass storage device is required. The TTY-43 is used both as an input and hard-copy output device. It prints 132 characters per print line. If your printer has a shorter print line, it makes no difference. Even if you don't have a hard-copy device, you can still use DATBAS for CRT output.

DATBAS programs are written in Business Basic, an interpreter sold by MicroWorks, Inc., of Cincinnati OH. The BASIC supports two commands used extensively in the DATBAS programs: CURSOR, which allows placement of the cursor anywhere on the CRT screen, and INPUT1, which is similar to the INPUT command that is used so often in BASIC, but it does not generate a carriage-return/line-feed after the user's input. Most BASICs support these commands in some form or another.

Input/output to mass storage devices is handled by strings. As a consequence, numeric information is read from the storage device via a string variable and, if required, converted to numeric information before use in arithmetic operations. This conversion process is accomplished through the CONVERT command, which is similar in its effects to the STR$ and VAL$ commands in other BASICs.

Additionally, Business Basic supports addressing individual elements in a string. Therefore, the code A$(1,3) addresses array elements 1 through 3 in string A$. The LEFT$, MID$ and RIGHT$ string functions accomplish the same things in other BASICs.

Definitions

A "data base" or "data file" is a collection of records relating to a particular subject. Its purpose is to provide organization to a body of facts about the subject. The basic building block of a data base is the "record."

A "record" contains information or known facts about some-

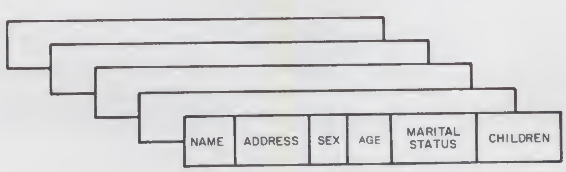


Fig. 1.

one or something included in the data base. Just as "records" represent the basic building blocks of a data base, the "data field" represents the basic building block of the "record."

Fig. 1 shows a pictorial summary of a sample data base that contains personnel information. The data base consists of five records. One record is devoted to each individual employee. Within each record are six data fields. These data fields are NAME, ADDRESS, SEX, AGE, MARITAL STATUS and CHILDREN. In the data base presented, actual information pertaining to the employee is entered for each data field. Therefore, a person's name is actually kept in the space designated NAME.

The Program

Having laid the groundwork, I will now discuss the program SCREEN. Remarks will center primarily on operational aspects of the program rather than on program code (REMark statements placed in the program are intended to serve this purpose).

The SCREEN program is used to create a "data entry screen," whose purpose is to allow the user to specify a record layout or format for data base records (that is, it allows the user to allocate record space for each data field). To accomplish this task, SCREEN divides the CRT display into labels and data fields. Labels are specified by the user and are used to identify information the user intends to be included in the record.

In the personnel file example, user-specified labels are: NAME, ADDRESS, SEX, AGE, MARITAL STATUS and CHILDREN. It should be noted that labels serve only as a prompting device at data entry time and are not stored as part of each record in the data base. To do so would unnecessarily waste storage space.

Data fields are used to hold the actual information or data entered by the user. The length of each data field is specified by the user. DATBAS programs assume fixed-length data fields, which means that the length of a given data field is assumed to be the same for all records. Therefore, if the user specifies the data field associated with the label NAME to be 20 characters in length, it is assumed to be 20 characters in length for all records in the data base.

As a result, you must be careful to ensure that the data field specified is large enough to hold the biggest entry into that field for any record in the data base. To specify 20 characters for NAME and then attempt to enter 30 characters will result in the last 10 characters being lost.

SCREEN program.

```
10  REM     A$   INPUT/OUTPUT STRING FOR DISK DRIVES -- SERVES
20  REM          ALSO TO NULL D$ AFTER USER LABEL INPUTS
30  REM     B$   HOLDS FILE NAME
40  REM     C$   UTILITY STRING FOR USER RESPONSES
50  REM     D$   USED TO HOLD USER LABEL INPUTS
60  REM     E$   HOLDS ALL LABEL INPUTS
70  REM    C()   COLUMN ADDRESS FOR CURSOR POSITIONING ON CRT
80  REM    F()   HOLDS LENGTH OF EACH DATA FIELD
90  REM    R()   ROW ADDRESS FOR CURSOR POSITIONING ON CRT
100  REM   L()   HOLDS LENGTH OF INDIVIDUAL LABEL FIELDS
110  REM   N()   HOLDS NUMERIC FIELD DESIGNATOR
120  REM   D     INITIALLY USED TO HOLD I/O DEVICE NUMBER
130  REM         LATER USED TO HOLD TOTAL LENGTH OF RECORD
140  REM   G     HOLDS CURRENT LENGTH OF SCREEN LINE.
150  REM         THAT IS, SUM OF LABEL AND DATA FILEDS + 4
160  REM   J     BEGINNING POSITION OF LABEL IN E$
170  REM   L     HOLDS NUMBER OF LABELS TO BE ENTERED
180  REM   K     ENDING POSITION OF LABEL IN E$
190  REM   Q     HOLDS REMAINING DATA SPACE IN RECORD
200  REM   Z     HOLDS NUMBER OF LABELS ACTUALLY INPUTED
210  REM  B1     HOLDS NUMBER OF RECORDS PER 256 BYTE BLOCK
220  REM  L1     HOLDS LINE POSITION DATA ENTRY SCREEN TO BEGIN
230  REM  L2     HOLDS NUMBER OF REMAINING LINES ON CRT
240  DIM A$(256),B$(10),C$(10),D$(256),E$(512),C(30),F(30)
250  DIM L(30),N(30),R(30)
260  #"" : FOR I=1 TO 7 : #" " : NEXT I
270  #"                    DATA ENTRY SCREEN PROGRAM "
280  #"                         VERSION 1.01 "
290  #"                           04/10/79 "
300  REM BASIC REQUIRES THAT ALL STRINGS BE INITIALIZED BEFORE
310  REM FIRST USE.
320  A$=" " : FOR I=1 TO 8 : A$=A$+A$ : NEXT I
330  D$=A$ : B$=A$(1,10) : C$=B$ : E$=A$+D$
340  FOR I=1 TO 5 : #" " : NEXT I
350  INPUT "Enter input/output device number (0 or 1) ",D
360  IF D<0 OR D>1 THEN #"Enter 0 or 1 " : GOTO 350
370  INPUT "Enter input screen file name ",B$
380  IF LEN(B$)<>6 THEN #"6 characters must be entered " : GOTO 370
390  B$=B$+".SE"+CHR$(0)
400  OPEN (0,E,B$,1,D,3)
410  IF E>1 THEN #"Open error ";E : GOTO 350
420  INPUT "Enter number of field labels ",L
430  IF L<1 OR L>30 THEN #"Label number must be 1 thru 30 " : GOTO 420
440  INPUT "Enter screen line labels are to begin (0-14) ",L1
450  IF L1<1 OR L1>14 THEN #"Line number incorrect " : GOTO 440
460  R=L1 : C(1)=0 : C=0 : D=0 : J=1
470  FOR I=1 TO 30 : N(I)=0 : NEXT I : REM THIS ZEROES NUMERIC FIELD DESIGNATOR
480  Z=0
490  FOR I=1 TO L
500    Q=256-D
510    #"Label ";%2I;I;"    Data space left is ";%3I;Q;" bytes"
520    L2=14-R
530    #"** Number of screen lines remaining is ";%2I;L2;" **"
540    INPUT "Enter screen label ",D$
550    L(I)=LEN(D$)
560    INPUT "Enter data field length ",F(I)
570    G=L(I)+F(I)+4
580    IF G>63 THEN #"Line too long ";G : D$=A$ : GOTO 540
590    INPUT "Numeric data field (Y/N) ",C$
600    IF C$="Y" OR C$="y" THEN N(I)=1
610    C(I)=C
620    K=J+L(I)-1
630    E$(J,K)=D$
640    D$=A$
650    J=K+1
660    C=C+G
670    IF C>63 THEN R=R+1
680    IF R>14 THEN #"Screen full, last entry ignored " : EXIT 760
690    IF C>63 THEN C(I)=0 : C=G
700    D=D+F(I)
710    IF D>256 THEN #"Data space full, last entry ignored "
720    IF D>256 THEN D=D-F(I) : EXIT 760
730    Z=Z+1
740    R(I)=R
750    NEXT I
760  B1=1
770  D$=A$ : REM NULL D$, USE IT TO NULL A$ LATER
780  REM DETERMINE HOW MANY RECORDS CAN BE PUT IN 256 BYTE BLOCK
790  B1=INT(256/D)
800  #"Blocking factor is ";B1
810  #"Data input screen will look as follows "
820  FOR I=1 TO 500 : # : NEXT I
830  REM CLEAR SCREEN WITH CONTROL L.  SHOW WHAT INPUT SCREEN
840  REM WILL LOOK LIKE WHEN PROGRAM FILEIT IS USED.
850  #"" : REM CONTROL L CLEARS SCREEN AND HOMES CURSOR
860  J=1
870  FOR I=1 TO Z
880    CURSOR R(I),C(1)
890    K=J+L(I)-1
900    REM DISPLAY ON CRT USER LABELS
910    #E$(J,K);
920    J=K+1
930    #" ";
940    FOR M=1 TO F(I)
950      REM SHOW NUMBER OF SPACES ALLOCATED FOR EACH DATA FIELD
960      #"x";
970      NEXT M
980   NEXT I
990  REM STORE AWAY ALL THE INFORMATION ENTERED
1000  REM   RECORD ZERO LENGTHS OF INDIVIDUAL LABELS AND DATA
1010  REM               FIELDS. FIELDS ARE ASSUMED TO BE 2 IN
1020  REM               LENGTH. LABELS LENGTHS IN POS 1-120.
1030  REM               DATA FIELD LENGTHS IN POS 121-240.
1040  REM               NUMBER OF LABELS ENTERED POS 241-242.
1050  REM               BEGINNING LINE FOR CRT DISPLAY POS 243-244
1060  REM               BLOCKING FACTOR POS 245-246
1070  REM               TOTAL LENGTH OF DATA RECORD POS 247-249
1080  FOR I=1 TO Z
1090    CONVERT L(I) TO C$(##)
1100    A$(I*2-1,I*2)=C$
1110    CONVERT F(I) TO C$(##)
1120    A$(119+I*2,120+I*2)=C$
1130    NEXT I
1140  CONVERT Z TO C$(##)
1150  A$(241,242)=C$
1160  CONVERT L1 TO C$(##)
1170  A$(243,244)=C$
1180  CONVERT B1 TO C$(##)
1190  A$(245,246)=C$
1200  CONVERT D TO C$(###)
1210  A$(247,249)=C$
1220  PUT (0,E,A$,0)
1230  REM D$ IS NOW USED TO NULL A$
1240  A$=D$
1250  REM STORE ALL LABELS IN RECORDS 1 AND 2.
1260  FOR I=1 TO 2
1270    A$(1,256)=E$(1*256-255,I*256)
1280    PUT (0,E,A$,I)
1290    NEXT I
1300  A$=D$
1310  REM STORE ROW AND COLUMN ADDRESSES FOR CURSOR POSITIONING
1320  REM THEY ARE STORED IN RECORD 3.  ROWS ARE STORED IN
1330  REM POS 1-120 (ASSUMED TO BE 2 DIGITS EACH).  COLUMNS ARE
1340  REM STORED IN POS 121-240 (ASSUMED TO BE 2 DIGITS EACH).
1350  FOR I=1 TO L
1360    CONVERT R(I) TO C$(##)
1370    A$(I*2-1,I*2)=C$
1380    CONVERT C(I) TO C$(##)
1390    A$(119+I*2,120+I*2)=C$
1400    NEXT I
1410  PUT (0,E,A$,3)
1420  A$=D$
1430  FOR I=1 TO L
1440    CONVERT N(I) TO C$(#)
1450    A$(I,I)=C$
1460    NEXT I
1470  PUT (0,E,A$,4) : REM THIS STORES NUMERIC FIELD DESIGNATORS
1480  CLOSE (0,E)
1490  CURSOR 15,0
1500  #"End of processing"
1510  END
```

```
Enter input/output device number (0 or 1) 0
Enter input screen file name PERSON
Enter number of field labels 6
Enter screen line labels are to begin (0-14) 0
Label  1    Data space left is 256 bytes
** Number of screen lines remaining is 14 **
Enter screen label NAME
Enter data field length 20
Numeric data field (Y/N) N
Label  2    Data space left is 236 bytes
** Number of screen lines remaining is 14 **
Enter screen label ADDRESS
Enter data field length 30
Numeric data field (Y/N) N
Label  3    Data space left is 206 bytes
** Number of screen lines remaining is 13 **
Enter screen label SEX
Enter data field length 1
Numeric data field (Y/N) N
Label  4    Data space left is 205 bytes
** Number of screen lines remaining is 13 **
Enter screen label AGE
Enter data field length 2
Numeric data field (Y/N) Y
Label  5    Data space left is 203 bytes
** Number of screen lines remaining is 13 **
Enter screen label MARITAL STATUS
Enter data field length 1
Numeric data field (Y/N) N
Label  6    Data space left is 202 bytes
** Number of screen lines remaining is 12 **
Enter screen label CHILDREN
Enter data field length 2
Numeric data field (Y/N) Y
```

Sample run.

To demonstrate SCREEN's use, a data entry screen will be created from the sample personnel file previously described. Although not much preparatory work is required for the present application, in other applications, some initial time spent in developing label names and data field lengths may save much more time later.

You will see in next month's follow-up article on FILEIT that label names are used quite extensively. Therefore, developing concise, meaningful labels at the outset will carry with it its own rewards in the manipulation of the data base. Making sure of data field lengths will avoid the disaster of having to reformat or, worse yet, reenter all records because a data field specified was too small to hold the information to be entered.

Normally, to effectively use SCREEN, initial preparation requires setting down the labels to be given data fields. The labels used in this case are those previously defined: NAME, ADDRESS, SEX, AGE, MARITAL STATUS and CHILDREN. After identifying labels have been developed, the amount of record space devoted to each data field must be determined. In this case NAME is assumed to be no longer than 20 characters; ADDRESS, 30 characters; SEX, one character; AGE, two characters; MARITAL STATUS, one character; and CHILDREN, two characters.

The dialogue between the user and the computer is shown in the sample run. The process shown is representative of steps taken to input labels and to define data fields in a record. Since the process is relatively to the point, no further discussion on the procedure is necessary. However, a summary statement of some of SCREEN's limitations are in order:

- CRT lines that are available for user display purposes are lines (0–14). Line 15 of the CRT is used for messages to the user.
- Number of label and data fields is limited to 30.
- A label and its associated data fields cannot be longer than 63.

A sample of SCREEN's output is shown in Fig. 2. The small x's following each label represent the amount of space the user has allotted to each data field. These x's will also be used in the FILEIT program to remind the user of the data field's length.

Before closing, I should mention one final aspect of SCREEN that relates to the blocking of records.* SCREEN automatically calculates the number of records that can be grouped together to fill a 256 byte block.

As a result, if the user's records are only 64 bytes in length, SCREEN will generate a blocking factor of four. That means four records will be input or output to the mass storage device with each read or write. The blocking factor will be used by the FILEIT program as part of the data base creation and maintenance process. ■

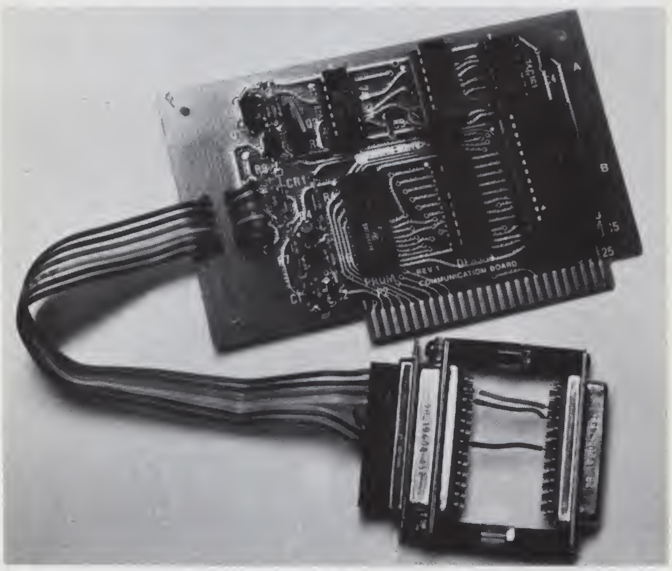
```
NAME xxxxxxxxxxxxxxxxxxxx
ADDRESS xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  SEX x  AGE xx
MARITAL STATUS x  CHILDREN xx
```

Fig. 2.

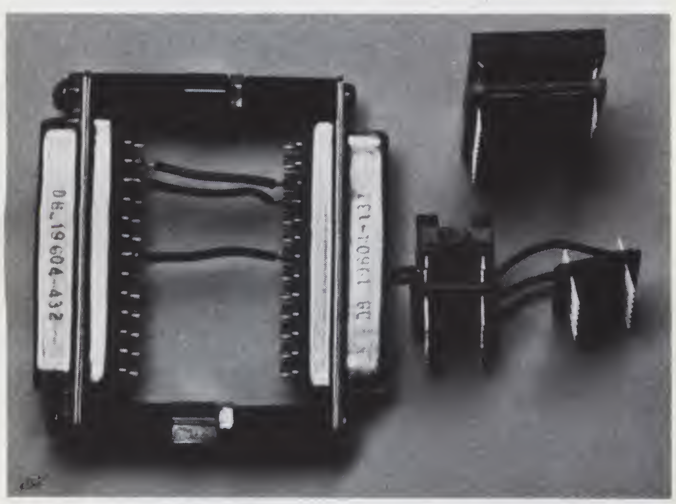
*See "Data-File Creation Program," *Microcomputing*, July 1979, p. 44.

Fast Apple Peripherals

How to interface high-speed serial printers to the Apple II microcomputer.



Modified Communications Interface Card with all adapters installed.



The three adapters.

Bruce S. Chamberlain
917 Delmas Ave.
San Jose CA 95125

When I got my Apple II, I wanted to use it for word processing, among other things, so I needed a printer. I picked the Heathkit H14 printer because it printed upper and lowercase letters, was fast and inexpensive ($645, kit). There was one catch. It had an RS-232 interface and needed handshaking to run at 4800 baud full speed.

Handshaking allows the Apple and the printer to send control signals to each other. Unfortunately, Apple does not make a serial card with handshaking. All is not lost, however. With slight modification, you can add handshaking to Apple's Communication Card or High-Speed Serial Interface Card.

Board Modification

Since I wanted handshaking to be in hardware, I used the Apple Communications Interface Card. The communications card has a UART, the 6850 ACIA, which controls the transmission of characters. To modify the communications card, you connect the clear-to-send line of the 6850 (pin 24) to the busy signal of the printer.

The clear-to-send line is active low, and the printer busy signal is active high. By active, I mean true, i.e., if the signal is high, the printer is busy and it is not clear to send. All that is needed to connect them together is a non-inverting buffer to handle the differing voltage levels. Since I was receiving nothing from the printer, I used the receive buffer.

However, the 2N3904 transistor, which forms the receive buffer, inverted the signal, so I changed it to a 2N3906 with its collector to ground (Fig. 1). Thus, when the printer busy signal goes high, signaling that the printer buffer is full, the transistor is turned off, and the clear-to-send pin goes high stopping output of characters to the printer. The printer has a 256 character buffer, which stores the characters it receives until they can be printed.

By using two adapters, I did not have to cut any traces on the card. The first adapter is a 24-pin socket and a 24-pin platform soldered together with pin 2 of the platform connected to pin 24 of the socket, and pin 2 of the socket connected to pin 24 of the platform. All the other pins are soldered to the corresponding number, for example, platform pin 1 to socket pin 1 and platform pin 3 to socket pin 3 (Fig. 2).

The second adapter is a male 25-pin "D" connector connected to a female 25-pin "D" connector with pins 7 connected to each

other, pin 2 of the male connector to pin 3 of the female connector and pin 3 of the male connector to pin 4 of the female connector (Fig. 3).

This gave me handshaking but with only a transmission rate of 300 baud, which is the stock transmission rate of the communications card. Fig. 4 shows how to rewire the counters to get a transmission rate of 4800 baud. Use another adapter consisting of two 16-pin platforms and a 16-pin socket. Solder the socket on one of the platforms with pins 5, 6 and 8 to pin 8 of the platform. Wires from pins 15 and 11 are soldered to pins 15 and 11, respectively, of the second platform.

All other pins of the socket are soldered to their respective platform pins. Place the platform and socket assembly in socket A2 of the communications card and the platform in socket A1. Replace the 74C161 in A2 and save the other 74C161 for some other project. This gives a transmission rate of 4800 baud.

After plugging the modified board into my Apple I/O slot #2 and attaching the printer, I tried it out. It worked the first time.

Software Modification

With the help of Danny Lambert, who sold me the Text Editor for my Apple, I modified the Text Editor to work with the communications card. It had been written to work with the Parallel Printer Card. Now I have a word-processing system. It has spoiled me.

(Note: The communications card does not output a line feed after each carriage return. The H14 printer has an automatic line-feed option, which can be selected by flipping a switch in the printer. Without this switch it would print on the same line over and over again. Fortunately, the Text Editor supplies all of the control characters for printing. Listing a program works fine, but any program that uses the tab feature in the output will not work. The communications card has no tab-handling routine.)

Apple's Communications Interface Card Manual has a machine-language print routine written by Wendell Sanders. After typing it into the Apple and correcting two spots where I had typed "B" instead of "8," it ran fine. Now I had line feeds and TABS, but I wanted to list my programs the same way the word processor outputs text—a page at a time, with my choice of how many lines per page.

My Apple II system.

Studying the listing for the print routine, I tried to figure out how it worked. I understood very little. Most of the lines referred to various locations the Apple monitor uses to store information to run the system.

Finally, I zeroed in on two lines near the end, labeled "20 msec delay after line feed." I did not need this since I had handshaking that stopped the output when the print buffer was almost full. This was the ideal location to count the number of lines and output a form feed when it had counted a certain number of lines. I removed these two lines and inserted a jump to a subroutine, which counted lines and output a form feed after 24 lines had been printed.

I ran it, and it worked fine. I could now tear apart the sheets without having the top half of a line on one sheet and the bottom half of the line on the other sheet. But I still could not punch them and put them in a ring binder, because I had no left margin. I added a small loop to

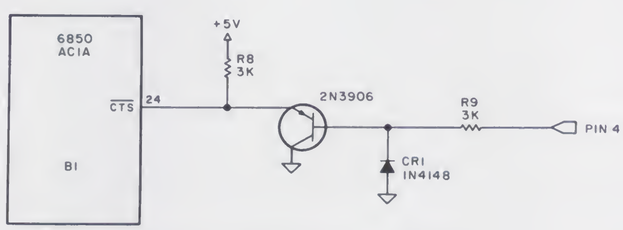


Fig. 1. Transistor modification.

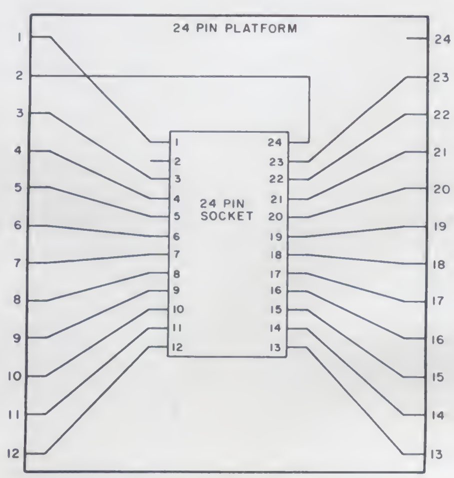


Fig. 2. Pin configurations.

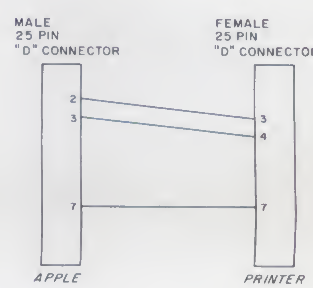
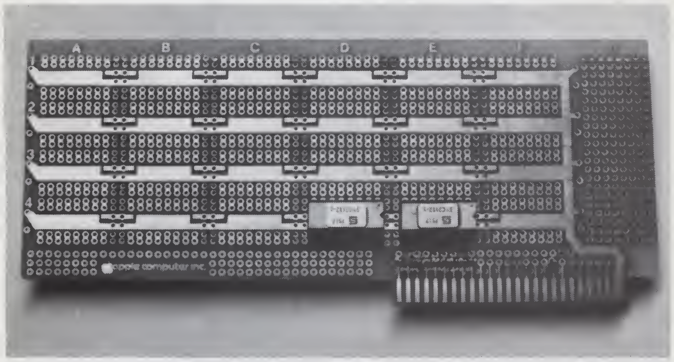
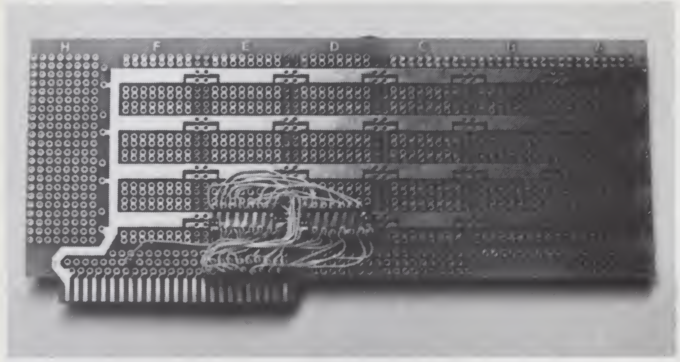


Fig. 3. Computer to printer pinouts.



Component side of 256 byte RAM card.



Wire-wrap side of 256 byte RAM card.

the subroutine that output ten blanks before each line was printed. I used the program to output a disassembler listing of itself. This time I could tear apart the pages of the listing, punch them and put them in a ring binder.

When I tried to load a program from the disk, so I could print a listing of it, I discovered that the DOS (disk operating system) had disappeared. John Crossly and Randy Peterson, the two men who man Apple's hot line, (408) 996-9868, told me that I could only use 300 to 3CF hex for the program, as AppleSoft and DOS used 3D0 to 3FF hex for storage of pointers. This fact is not mentioned in the Apple manuals that I have.

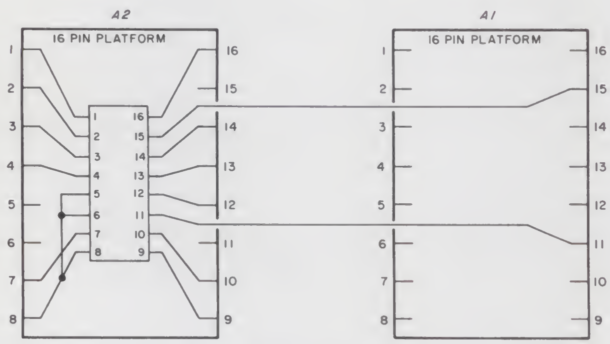


Fig. 4. Rewiring the counters for speed.

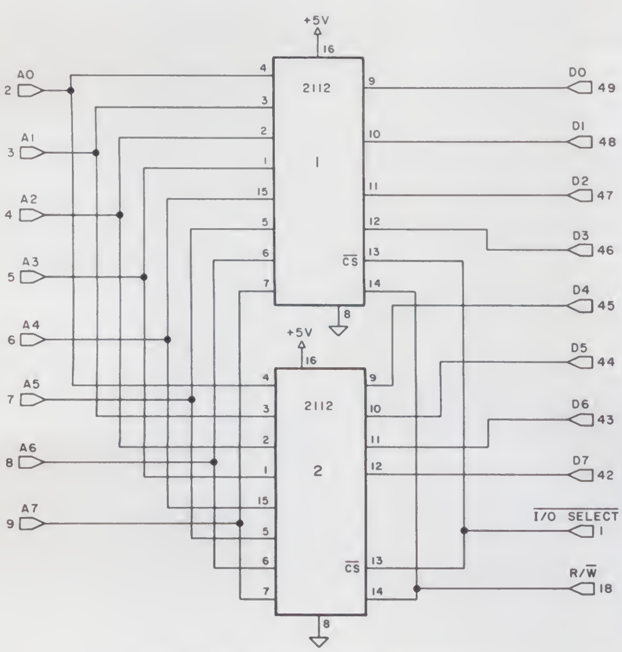


Fig. 5. Building a RAM board.

I shortened the program as much as I could but could not get it short enough. I did get it short enough so it did not destroy the DOS pointers, but not enough to save the AppleSoft ones. I could load programs but could not list the AppleSoft ones.

I decided to try and remove the part that gives me TABS, since I did not need them for listing. After some trial and error, I was able to shorten the program to CF bytes. Now I could print listings of assembly language, Integer BASIC and AppleSoft programs, which could be torn apart, punched and put in a ring binder.

All of this assembly-language rewriting was a chore. Looking for an easier way to write assembly language, I got a disk-based Editor/Assembler. This made assembly-language programming a breeze. There was one catch. It would not print out listings; the print routine would not work with it.

To run the print routine, it was necessary to start it from outside the Assembler/Editor, since the Assembler/Editor has no command to start a program. Then when I ran the Assembler/Editor, it changed the output pointers the print routine had set. To run the Assembler/Editor, you had to shut off the printer.

I could get a listing by using PR#2, which is a valid command in the Assembler/Editor. But since the communications card did not provide TABS, the listing was hard to read.

Communications Card Driver

All of this experimenting was teaching me more about how the Apple works. I decided to try and write a complete program that would replace the firmware on the communications card.

First, so I could call the program with a PR# command, I built a 256 byte RAM board. Since the Apple does address decoding for 256 bytes of memory at each peripheral I/O slot, I just put two 2112 RAM chips on a hobby card. The 2112 is a 256 by 4 static RAM IC. Two of them give 256 bytes of RAM. I did not need to use any buffers. I merely attached the address and data pins of the ICs to the appropriate hobby-card pins.

The chip select pin (CS) was hooked to I/O select, which goes low when CNXX is addressed, "N" being the slot number. The read/write pins of the RAM were attached to the hobby card read/write pin (Fig. 5). Putting the card in slot #4 and using the programmers aid ROM, I ran the test to check memory at locations C400 to C4FF and found no errors.

Using the Assembler/Editor to write the program and assembling it in my custom RAM space, I finally came up with a version that worked with the Assembler/Editor. See Listing 1.

The comments in the listing explain how it works. The middle part, lines 59 to 97, are from Wendell Sanders' print routine shown in Apple's Communica-

STOCK
CONTINUOUS
CHECKS & STATEMENTS

PEACH TREE** USERS

Now you can order as few as 500 checks for your system - and get them in as little as 9 days!

*Low Minimum Order *Fast Delivery

We have forms for most systems-including the TRS-80*

The Best Choice for the Small Computer User.

SEND FOR YOUR FREE SAMPLES TODAY!

FREIGHT INCLUDED

* Personal Checks
* Payables Checks
* Payroll Checks
* Combination Checks
* Statements
* Invoices
* Small Minimum Orders
* Fast Delivery

Checks To-Go ✓58

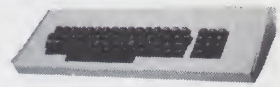
NATIONWIDE SERVICE

* Registered Trademark of Tandy Corp.

**Registered Trademark of Retail Sciences, Inc.

8384 Hercules St., La Mesa, CA 92041, (714) 460-4975

DATANETICS KEYBOARD ✓169



Has 75 keys including upper-lower case letters, standard numbers and symbols, plus separate ten key numerical set. ASCII encoded using 7400, 7404, 2/7408, 7410, 7432, 2/7493, NE556, and EA20106. +5 and −12 VDC required. Keyboard assembled. Metal enclosure unassembled with beige top, unpainted sides, brackets, rubber feet, 5' cable with PCB connector, and 16-pin dip plug. Pinout info. included. 20x7¾x2¾", 10 lbs. sh., NEW, **$89.95**

IC's — guaranteed no rejects. qty. limited.

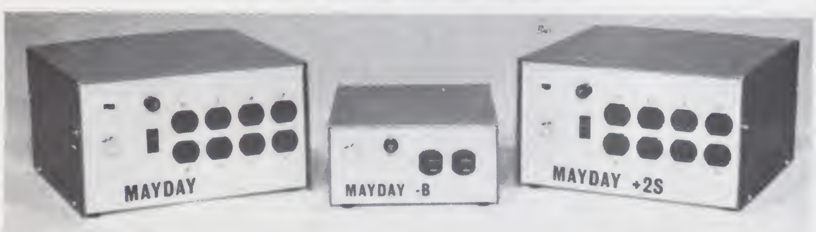
C8008-1 CPU	$10.95	2716 EPROM 5v	$27.00
8080A CPU	6.95	MC14411 b-gen.	9.95
P8085 CPU	16.95	3341 FIFO $5.95	4/$21.
8205 decoder	2.75	AY5-1013 UART	4.50
8212 8-bit I/O	2.50	2N3055 NPN trans.	.60
2102 RAM $1.40	4/$4.	TIP2955 PNP	.80
2111 RAM 3.60	4/$13.	MPSA14 NPN darl.	5/$1.
2114 RAM 5.95	4/$21.	2N6346 unijunc	.90
2708 EPROM erased-tested, $6.50	4/$22.		

ROTRON SPRITE FAN, 3-1/8x3-1/8x 1-5/8, 115 VAC, #SU2A3, NEW **$12.95**

$5.00 Minimum Order — Prices F.O.B. Lima, O.
Allow for shipping — We accept VISA or Mastercharge
Ask for our 36 page FREE CATALOG. • Dept. K
Electronic Surplus Since 1947 Phone 419/227-6573

FAIR RADIO SALES
1016 E. EUREKA • Box 1105 • LIMA, OHIO • 45802

introducing... **MAYDAY**™



The Uninterruptable Power Supply that....

- Prevents loss of memory when power "blackouts" occur.
- Prevents loss of memory or disk I/O errors when "brownouts" occur.
- Provides time to continue operating complete system with disk drives to finish operations or store information.
- Will handle most mini/micro computers with power consumption up to 250 watts.
- Designed and developed using complete TRS-80 System.

Protect your Time and Investment

For price list and detailed specifications, contact your nearest Sun-Technology distributor or call direct to:

Sun - Technology, Inc. ✓152
Box 210
New Durham, New Hampshire 03855 **(603) 859-7110**
(Manufacturing high technology products since 1970)

BPS SOFTWARE

FOR OHIO SCIENTIFIC USERS

BPSort

- **MACHINE CODE** for high-speed sorting
- **BASIC** language program for establishing sort parameters
- Files of any length can be sorted
- Floppies and hard disks are supported
- Up to five keys can be specified
- Ascending and descending keys are processed simultaneously
- "Header" records can be skipped if present
- Requires **OS-65U**

BPSort $124.00

EASY TO USE AND EXTREMELY FAST. A FILE OF **TWENTY-THOUSAND** BYTES CAN BE SORTED IN **LESS** THAN **10 SECONDS.**

Order C.O.D. Master Charge or Visa from

BPS
322 West 57th Street
New York, NY 10019
Telephone: 212-765-0815 ✓168

tions Interface Card Manual. Comments on this part of the program can be found in the listing of the routine in Apple's manual.

Wrap-up

Apple's High-Speed Serial Interface Card can be modified for handshaking. Just change two wires and use a short program to detect when the printer is busy. For details, call Apple's hot line.

I prefer the communications card with hardware handshaking. Programs that output print, which usually contain formatting commands, will work with the communications card without modification. If you use the serial card, you will have to add the handshaking software to each program.

This wouldn't be too much trouble if you wrote the program, but what if you bought one? Even an experienced programmer would need time to figure out the program so the handshaking software could be added. Therefore, I recommend using the communications card.

I now have a new Text Editor, which I used to write this article. It works fine with my setup. If I had the serial card setup, I would have had to modify the program for handshaking. This would be extremely difficult, since I have no source listing.

In conclusion, I am happy with my system. I have a versatile word-processing system, which only costs $4000. In addition, I have all the other features the Apple II offers. While this modification has proven satisfactory to me, I must warn you that *any* modification to Apple's products voids the accompanying warranties. ■

Listing 1. Communications Card Driver.

```
000:                *******************************
0000:               *                             *
0000:               *    COMMUNICATIONS  CARD     *
0000:               *                             *
0000:               *          DRIVER             *
0000:               *                             *
0000:               *    BRUCE S. CHAMBERLAIN     *
0000:               *                             *
0000:               *******************************
C400:                              ORG $C400
C400:                              OBJ $6000
C400:               CH             EQU $24
C400:               STATUS         EQU $C0FE
C400:               STAT           EQU $7F8
C400:               DATA           EQU $C0FF
C400:               MARGIN         EQU $0A
C400:               LINE1          EQU $18
C400:               COUT1          EQU $FDF0
C400:               *
C400:               *  ZERO COUNTERS
C400:               *
C400: A9 00                        LDA #$00
C402: 8D 26 C4                     STA LINECO
C405: 8D 27 C4                     STA MARCO
C408: 8D 25 C4                     STA COLCNT
C40B:               *
C40B:               *  INITIALIZE ACIA
C40B:               *
C40B: A9 03                        LDA #$3
C40D: 8D FE C0                     STA STATUS
C410: A9 11                        LDA #$11
C412: 8D FE C0                     STA STATUS
C415:               *
C415:               *  SET COLUMN WIDTH
C415:               *
C415: A9 28                        LDA #$28
C417: 8D 24 C4                     STA CWIDTH
C41A:               *
C41A:               *  SET OUTPUT POINTER
C41A:               *
C41A: A9 28                        LDA #TTOUT
C41C: 85 36                        STA $36
C41E: A9 C4                        LDA #TTOUT/256
C420: 8D F8 07                     STA STAT
C423:               *
C423:               *  RETURN TO CALLER
C423:               *
C423: 60            RTS1           RTS
C424:               *
C424:               *  COUNTERS & COLUMN WIDTH
C424:               *
C424: FF            CWIDTH         DFB $FF
C425: FF            COLCNT         DFB $FF
C426: FF            LINECO         DFB $FF
C427: FF            MARCO          DFB $FF
C428:               *
C428:               *  CHARACTER ENTRY POINT
C428:               *
C428: 48            TTOUT          PHA
C429: 48                           PHA
C42A: AD 25 C4      TTOUT2         LDA COLCNT
C42D: C5 24                        CMP CH
C42F: 68                           PLA
C430: B0 03                        BCS TEST
C432: 48                           PHA
C433: A9 A0                        LDA #$A0
C435: 2C 23 C4      TEST           BIT RTS1
C438: F0 03                        BEQ PRINTIT
C43A: EE 25 C4                     INC COLCNT
C43D: 08            PRINTIT        PHP
C43E: 20 6E C4                     JSR DOCHAR
C441: 28                           PLP
C442: 68                           PLA
C443: 48                           PHA
C444: 90 E4                        BCC TTOUT2
C446: E6 24                        INC CH
C448: 49 0D                        EOR #$0D
C44A: 0A                           ASL A
C44B: D0 0B                        BNE FINISH
C44D: 8D 25 C4                     STA COLCNT
C450: A9 8A                        LDA #$8A
C452: 20 6E C4                     JSR DOCHAR
C455:               *
C455:               *  JUMP TO SUBROUTINE AFTER EACH LINE
C455:               *
C455: 20 7B C4                     JSR LINE
C458: A9 00         FINISH         LDA #$0
C45A: 85 24                        STA CH
C45C: AD 25 C4      FINISH1        LDA COLCNT
C45F: F0 09                        BEQ SETCH
C461: ED 24 C4                     SBC CWIDTH
C464: E9 F7                        SBC #$F7
C466: 90 04                        BCC RETURN
C468: 69 1F                        ADC #$1F
C46A: 85 24         SETCH          STA CH
C46C: 68            RETURN         PLA
C46D: 60                           RTS
C46E:               *
C46E:               *  CHARACTER OUTPUT POUTINE
C46E:               *
C46E: 48            DOCHAR         PHA
C46F: AD FE C0      SEROUT         LDA STATUS
C472: 29 02                        AND #$2
C474:               *
C474:               *  WAITS FOR CLEAR-TO-SEND
C474:               *
C474: F0 F9                        BEQ SEROUT
C476: 68                           PLA
C477:               *
C477:               *  OUTPUTS CHARACTER
C477:               *
C477: 8D FF C0                     STA DATA
C47A: 60                           RTS
C47B:               *
C47B:               *  LINE COUNT ROUTINE
C47B:               *
C47B: EE 26 C4      LINE           INC LINECO
C47E: 48                           PHA
C47F: 08                           PHP
C480: AD 26 C4                     LDA LINECO
C483:               *
C483:               *  CHECKS FOR 24 LINES
C483:               *
C483: C9 18                        CMP #LINE1
C485:               *
C485:               *  BRANCH IF NOT 24 LINES
C485:               *
C485: D0 0A                        BNE NOFORM
C487:               *
C487:               *  RESETS LINE COUNTER
C487:               *
C487: A9 00                        LDA #$00
C489: 8D 26 C4                     STA LINECO
C48C:               *
C48C:               *  OUTPUTS FORM FEED
C48C:               *
C48C: A9 8C                        LDA #$8C
C48E: 20 6E C4                     JSR DOCHAR
C491:               *
C491:               *  10 SPACE MARGIN ROUTINE
C491:               *
C491: A9 A0         NOFORM         LDA #$A0
C493: 20 6E C4                     JSR DOCHAR
C496: EE 27 C4                     INC MARCO
C499: AD 27 C4                     LDA MARCO
C49C:               *
C49C:               *  CHECK FOR 10 SPACES
C49C:               *
C49C: C9 0A                        CMP #MARGIN
C49E:               *
C49E:               *  BRANCH IF NOT 10 SPACES
C49E:               *
C49E: D0 F1                        BNE NOFORM
C4A0:               *
C4A0:               *  RESETS MARGIN COUNTER
C4A0:               *
C4A0: A9 00                        LDA #$00
C4A2: 8D 27 C4                     STA MARCO
C4A5: 28                           PLP
C4A6: 68                           PLA
C4A7: 60                           RTS
```

$159.95 OSBORNE BUSINESS SOFTWARE

General Ledger	—$ 59.95
Accounts Payable & Accounts Receivable	— " "
Payroll & Job Cost	— " "
All Three—Complete	—$159.95

format - 8" single density CP/M diskette
requires CP/M & CBASIC - 2
manuals - $20. each

VANDATA ✓113

17541 stone avenue n. • seattle, washington 98133

(206) 542-8370

For Business Systems Software Programmers Only!!
Machine Language ISAM FOR TRS-80* Models I and II
ACCESS TIME ½ SECOND!!

This machine language **ISAM ROUTINE** provides file access sophistication required to implement complex business software, and is completely compatible with TRS-80* BASIC.

Specifications: based on 1000 record file, key length of 6, data length of 64

Access time of ½ second
Best add/delete time of 1 second
Average add/delete time of 2 seconds
Worst case add/delete time of 5 seconds
Keys up to 25 characters
Data up to 255
Provides next highest key upon each file access.
Preset range of allowable key values, set upon file creation up to 4 files held open concurrently.
No special utilities needed to "Reclaim" used record space.
Model I Routine occupies 5 K of user space (loads at top of user memory)
Model II Routine occupies 6 K of user space (loads at top of user memory)
EOF, file full, empty, key not found, INVALID key value and all **TRS DOS*** error codes supported.

TRS-80 Model I (32 K + 48 K only) $160.00
TRS-80 Model II (64 K only) $170.00

Includes: All necessary documentation, file creation program, file inspection program, machine language ISAM routine and loader.

RCR INC

SYSTEMS SPECIALISTS
415 MILLBURY STREET
WORCESTER, MA 01607
✓171 **(617) 755-8134**

**TRS-80 and TRSDOS are trademarks of Tandy Corporation, which has no relation to RCR Inc..*

HOME CONTROL SYSTEM FOR TRS-80*

Remote control through your existing house wiring. Control lights and appliances with your BASIC programs. **Schedule** the coffee pot, TV, radio, wake-up lighting, bathroom heater, porch lights, etc. with the ease of a carefully thought out scheduler program. Our ultrasonic transmitter connects with your AUX plug, and allows your computer to control the increasingly popular, high quality, BSR X-10 control system, made by BSR (USA) LTD. Our system contains everything your computer needs to control up to 16 of the switch models listed below. Try the system for 14 days and, if not completely satisfied, return it undamaged for a prompt refund.

HC-PTR2 System includes the BSR Ultrasonic Command Console, our transmitter, relocatable machine language handler, BASIC demo program, and user manual. Also the Timing and Schedule program, which gives a screen display of day, date, and time in large characters and a system for easily setting up and updating a large schedule of repeating and special switching events. Provisions are included for fast save and reload of the schedule.

Requires 16K Level II $149.00
BSR Appliance Module, Lamp Module, or Wall Switch .... each $14.95
Buzzer Alarm — for a programmable alarm clock! Plugs into Appliance Module, operates on house current $14.95

All software is on cassette. Deduct $30 if you already have the Ultrasonic Command Console. If the transmitter malfunctions within 90 days of purchase, mail it to us, and we will repair it free of charge, unless abuse is apparent. Thereafter an $8 flat fee must be included. The fee may be changed without notice. BSR equipment is provided with a one year limited warranty by BSR (USA) LTD.

Arizona residents add 5% for tax.
Foreign orders add 10%.

SEND CHECK OR MONEY ORDER TO:
SOFTRONICS COMPUTER SERVICES
P.O. BOX 1465 • MESA, ARIZONA 85201 ✓60

*TRS-80 is a trademark of Tandy Corporation.

---

apple II* SOFTWARE Bonanza!

HOME FINANCE PROGRAM. The most complete and concise home budgeting program for the Apple II. Simply laid out and easy to use.

- 175 entries/twelve catagories per month or year.
- Month-to-date, year-to-date summaries.
- Classifies tax deductible expenditures.
- Balances and reconciles checkbook, etc.

A household necessity! Includes 30 page manual. **PRICE: $39.95**

OMNIBUS BANKING & FINANCIAL PAK. A program designed to aid businesses and consumers in long or short range financial planning.

- Examines investments, savings and annuities; mortgages and loans; depreciation and amortization schedules and much more.

A must for the capital conscious person! **PRICE: $59.95**

APPLE II TEXT PROCESSING SYSTEM. A true text editor and assembler. Create and edit Integer, Applesoft* and Assembly language programs!

- Cumbersome line numbers are no longer needed.
- Basic programs can be converted into text files, edited, then converted back.
- Uses all 56 standard 6502 opcode mnemonics, plus 6 additional pseudo opcodes.

Includes 43 page manual. **PRICE: $55.00**

GAME DISK

TRY YOUR LUCK! Slots, roulette, craps and blackjack. (All odds listed).

DIET. Individually charts a total weight loss program and schedule.

BIORHYTHM. Separate critical day listing with graph.

LONGEVITY. Based on medical statistics. How long will you live? Uncanny!

Requires Applesoft. **PRICE: $39.95**

*Apple II and Applesoft are trademarks of Apple Computers Inc. ✓61

MC/VISA

Digital Marketing
2670 Cherry Lane
Walnut Creek, CA 94596
(415) 938-2880

Dealer inquires welcomed

$10 Classy Chassis

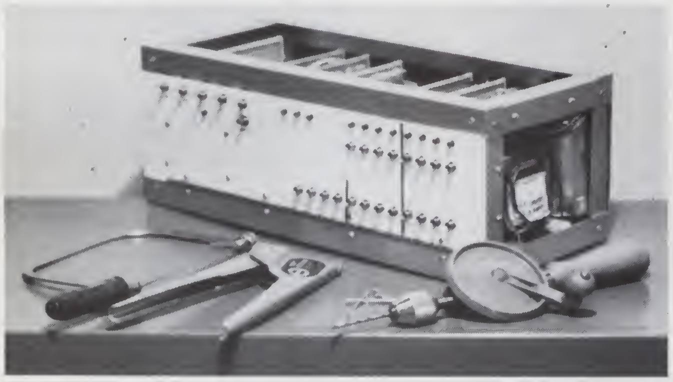
A metal frame for your micro can be one of its most expensive components. Make your own and save a bundle.

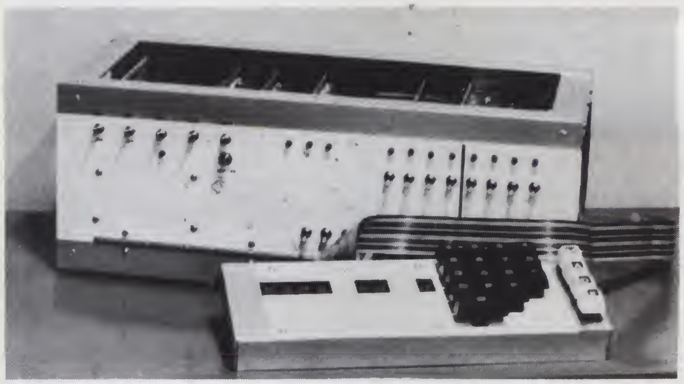
John Gledhill
678 Washington Ave., #4
Yuba City CA 95991

Face it. Computers built on wooden blocks are shabby; and commercial cabinets are expensive, even though they are rarely just the right size. Here's a chassis that's ideal for that home-brew computer or peripheral you've been thinking about.

It's basically a box framework made of 1 x 1/16 inch aluminum angle and pop rivets, available at any hardware store. Tools required are hacksaw (or coping saw), ⅛ inch drill and pop riveter. The aluminum is easy to cut and drill, and assembly is simple. The finished product is a strong, lightweight, professional-quality chassis that does justice to the time and effort needed to build a computer.

Eight identical corners make up the box, as shown in the photographs. Twenty-four pop rivets and ten feet or more (depending on dimensions) of aluminum angle are all you'll need. It will cost about $10. ■





My chassis is 16 x 6 x 7 inches. It has some extra members for mounting 22-pin dual-edge connectors and a power supply. Make sure to provide room for future expansion!

URGENT

Deadline May 7, 1980

MAIL NOW

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 105 TITUSVILLE, FLORIDA 32780

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTER SHOPPER
P.O. Box F15
Titusville, Florida 32780

LIST OF CATEGORIES

COMPUTERS
133 Borroughs Systems
136 Data General Systems
138 Data General, Software, Peripheral
142 Datapoint Systems
144 Datapoint Software, Peripheral
148 DEC Systems For Sale
149 DEC Systems Wanted
150 DEC Software, peripheral
154 IBM Systems For Sale
155 IBM Systems Wanted
156 IBM. Software, Peripheral
160 NCR Systems
162 NCR. Software, peripheral
166 Mics (Hardware & Software)

MICRO COMPUTERS
106 Apple Computers For Sale
107 Apple Computers Wanted
108 Apple, Software, Peripheral
112 Northstar Computers
114 Northstar, Software, Peripheral
117 Ohio Scientific
119 Ohio Scientific, Software Peripheral
122 PET Computers
124 PET Software, Peripheral
101 TRS-80 Computers For Sale
102 TRS-80 Computers Wanted
103 TRS-80, Software, Peripheral
126 Misc Microcomputers
128 Misc Micro, Software & Peripheral
130 S-100 Compatible Peripheral

PERIPHERAL & MISC. EQUIPMENT
170 Card Readers
172 Disk Drives
174 Printers
176 Punched Card Equipment
178 Tape Drives
180 CTR's
182 Terminals
184 Memory
186 Modems
188 Mics Equipment
190 Misc Large Systems
192 Misc Software
194 Misc Accessories & Supplies

EMPLOYMENT
196 Help Wanted
198 Positions Wanted
199 Other

FREE AD – FREE ISSUE

You may run your first classified ad FREE under NO obligation and receive a FREE issue too. Later you'll receive an invoice for a **half price**, $5 subscription, 12 issues/year. If you are not 100% satisfied with COMPUTER SHOPPER just write cancel on the invoice and the issues will stop coming, BUT you still keep that first issue as a free gift.

Print clearly or type your ad in the space below.

If you need additional space use a separate piece of paper and enclose it with this card in an envelope.

If you don't have an ad to send at this time, write in space below: SEND FREE AD CERTIFICATE. (Which you may use anytime during your subscription)

Category # ________ Your Name ____________________________

Address ____

FREE AD – FREE ISSUE

You may run your first classified ad FREE under NO obligation and receive a FREE issue too. Later you'll receive an invoice for a **half price**, $5 subscription, 12 issues/year. If you are not 100% satisfied with COMPUTER SHOPPER just write cancel on the invoice and the issues will stop coming, BUT you still keep that first issue as a free gift.

Print clearly or type your ad in the space below.

If you need additional space use a separate piece of paper and enclose it with this card in an envelope.

If you don't have an ad to send at this time, write in space below: SEND FREE AD CERTIFICATE. (Which you may use anytime during your subscription)

Category # ________ Your Name ______________________________

Address____

____

____

____

____

____

____

____

____

____

LIST OF CATEGORIES

COMPUTERS

133 Borroughs Systems
136 Data General Systems
138 Data General, Software, Peripheral
142 Datapoint Systems
144 Datapoint Software, Peripheral
148 DEC Systems For Sale
149 DEC Systems Wanted
150 DEC Software, peripheral
154 IBM Systems For Sale
155 IBM Systems Wanted
156 IBM, Software, Peripheral
160 NCR Systems
162 NCR, Software, peripheral
166 Mics (Hardware & Software)

MICRO COMPUTERS

106 Apple Computers For Sale
107 Apple Computers Wanted
108 Apple, Software, Peripheral
112 Northstar Computers
114 Northstar, Software, Peripheral
117 Ohio Scientific
119 Ohio Scientific, Software Peripheral
122 PET Computers
124 PET Software, Peripheral
101 TRS-80 Computers For Sale
102 TRS-80 Computers Wanted
103 TRS-80, Software, Peripheral
126 Misc Microcomputers
128 Misc Micro, Software & Peripheral
130 S-100 Compatible Peripheral

PERIPHERAL & MISC. EQUIPMENT

170 Card Readers
172 Disk Drives
174 Printers
176 Punched Card Equipment
178 Tape Drives
180 CTR's
182 Terminals
184 Memory
186 Modems
188 Mics Equipment
190 Misc Large Systems
192 Misc Software
194 Misc Accessories & Supplies

EMPLOYMENT

196 Help Wanted
198 Positions Wanted
199 Other

URGENT

Deadline May 7, 1980

MAIL NOW

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 105 TITUSVILLE, FLORIDA 32780

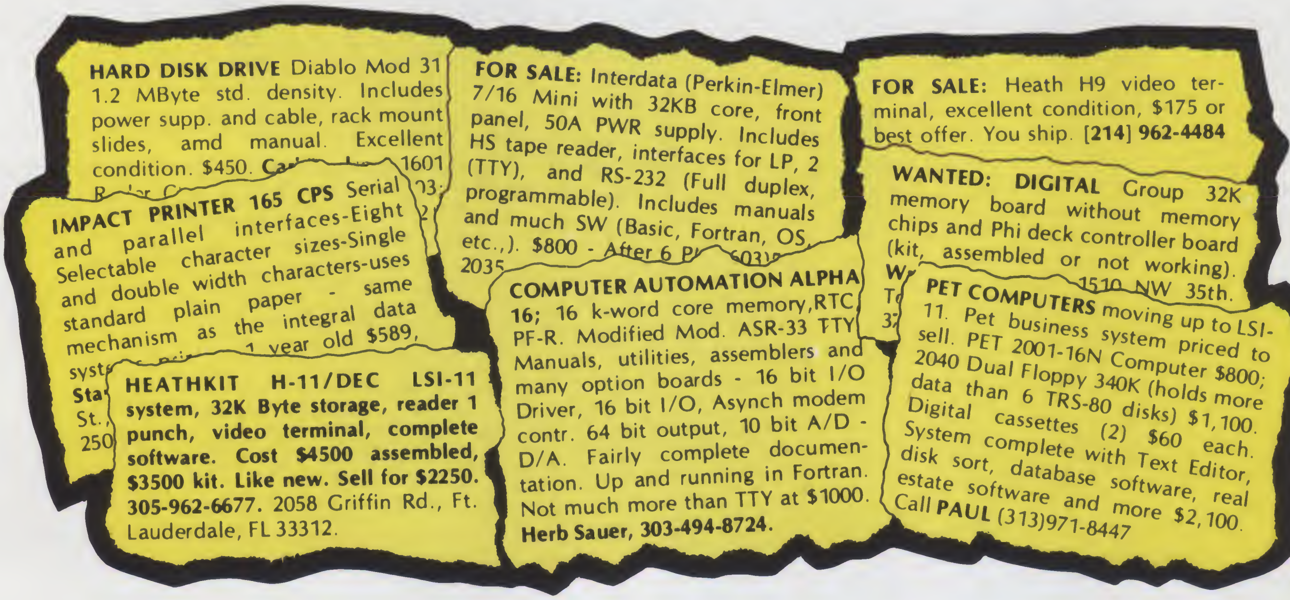
POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTER SHOPPER
P.O. Box F15
Titusville, Florida 32780

Attention

BARGAIN HUNTERS

Receive Hundreds of Classified Ads Like These Every Month



COMPUTER SHOPPER, the new buy, sell, and trade publication, is ready to help you with the latest information on personal, small business and large-system computers, accessories and software.

Each ad-packed issue is full of the bargains you are looking for. Included are ads from individuals throughout the United States who are selling their good, pre-owned equipment just so they can trade-up to new equipment coming on the market.

But, COMPUTER SHOPPER'S bargains won't be yours unless you subscribe. This useful, money-saving publication can become your way to communicate with other buyers, sellers, and traders all over the nation.

Whether you are a hobbyist or a part-time user, COMPUTER SHOPPER will put you in touch with the nationwide computer marketplace in time for you to take advantage of bargain opportunities.

Have something to sell? A COMPUTER SHOPPER subscriber probably wants to buy it.

Looking for a part, component or even a complete system? A COMPUTER SHOPPER subscriber probably wants to sell it.

COMPUTER SHOPPER is THE marketplace for anything in computers and is read by thousands of people who are ready to buy.

COMPUTER SHOPPER offers a unique format in which classified ads are categorized for fast location of specific items. Combining this with low individual ad rates — 10 cents a word — makes it the ideal place for buyers and sellers to communicate. And, its mix of individual, broker, and manufacturer ads enable subscribers to find what they want at the best price possible.

COMPUTER SHOPPER will work for you in other ways, too. If you are just thinking about getting into computers, it can help you learn product availability and prices before you make a decision. And, through the timely ads, COMPUTER SHOPPER will keep you abreast of changes in the market which could create bargain opportunities for you.

But COMPUTER SHOPPER cannot work for you unless you subscribe. Complete the subscription card today and put COMPUTER SHOPPER on your monthly reading list.

Want to look us over first? We'll give you your first issue FREE and then bill you for the next 12. If you are not convinced COMPUTER SHOPPER suits your needs, just write "cancel" on the invoice and return it.

And, to let COMPUTER SHOPPER start working *for you right now*, we'll also give you a FREE classified ad to sell your pre-owned equipment or to find equipment you want.

The FREE ad is yours whether you decide to subscribe or not with no obligation what-so-ever.

Don't miss another timely issue of COMPUTER SHOPPER. Send in the postpaid card today (give the other card to a friend) and let COMPUTER SHOPPER start working for you.

The Nationwide Marketplace for Computer Equipment

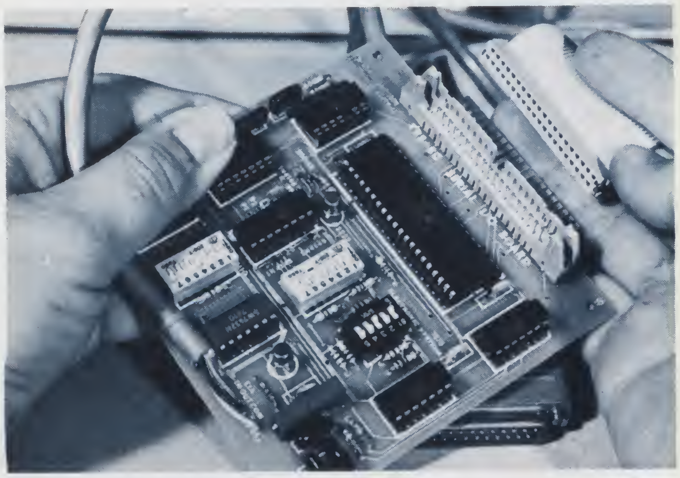
COMPUTER SHOPPER ✓36

P.O. Box F5 • Titusville, Florida 32780
Telephone 305-269-3211

✓*Reader Service—see page 257*

TRS-80 Serial I/O for Less

The expensive expansion interface is not needed with Electronic Systems' I/O board.



The Electronic Systems TRS-80 serial I/O board can be used to run printers, input terminal data or convert your TRS-80 to a micro-time-share business.

```
10 CLS:REM ELECTRONIC SYSTEMS PRINTER DRIVER
20 PRINT"28 BYTE LPRINT PROGRAM FOR SERIAL I/O BOARD"
30 PRINT"VERSION 1.0    1979 STEPHEN GIBSON"
40 PRINT"THIS PROGRAM PLACES A SHORT DRIVER ROUTINE AT THE TOP"
50 PRINT"OF YOUR MEMORY. YOU MUST RESERVE SPACE FOR IT IF YOU"
60 PRINT"HAVE NOT ALREADY DONE SO. SIMPLY RESET YOUR COMPUTER"
70 PRINT"AND ENTER THE APPROPRIATE MEMORY SIZE BELOW.":PRINT
80 PRINT"4K = 20450":PRINT"16K = 32738"
90 PRINT"32K = 49122":PRINT"48K = 65506"
100 PRINT:INPUT"HAVE YOU ENTERED ONE OF THE ABOVE SIZES Y/N";A$
110 IF A$="N" THEN PRINT"*** RESET AND DO SO NOW ***":END
120 INPUT"WHAT SIZE ABOVE DID YOU ENTER ";M
130 M=M+1:H=INT(M/256):L=M-256*H
140 POKE 16422,L: POKE 16423,H: POKE 16553,255
150 IF M>32767 THEN M=M-65536
160 FOR S = 0 TO 27 :READ G : POKE M+S,G :NEXT
170 INPUT"DO YOU WANT LINE FEEDS AFTER CARRAGE RETURNS Y/N";A$
180 IF A$="Y" THEN 190 ELSE POKE M+24,201
190 PRINT "DONE!" : END
200 DATA 121,254,13,40,3,254,32,216,58,232,55,230,36,254,36
210 DATA 32,247,121,50,232,55,254,13,192,14,10,24,236
```

Listing 1. A simple printer routine.

Stephen Gibson
PO Box 38386
Los Angeles CA 90038

Finding serial I/O for a Radio Shack TRS-80 has, until recently, been quite a hassle. While it has been relatively easy to obtain a serial-output device for printers, the availability of an inexpensive input/output device has been sorely lacking. True, Radio Shack does sell an RS-232 serial I/O board for which you can pay nearly $100 and simply drop it into their $299 expansion interface. Simple? . . . yes.

Now along comes Electronic Systems, PO Box 21638, San Jose CA 95151, that innovative and intrepid group of young lads whose ads you might have seen hawking nifty tape interfaces, modems, memory, TVTs and the like, all at reasonable prices . . . largely because you must assemble these units yourself. But recognizing the true appliance nature of the TRS-80 (nary a wire to solder), they've filled the TRS-80 serial I/O void and done all the soldering to boot. Their unit is available assembled and ready to plug in.

A kit version is offered. But unlike the universal serial I/O board the lads sell from their catalog, this new TRS-80 serial I/O has RS-232 capability, which is darned near a must if you intend to plug in a modem or terminal.

More important, you don't need an expansion interface to use it! Just plug it into the 40-pin connector on the back. Expansion-interface users can still plug in by using the convenient extension connector on the board. This neat feature, designed by Bob Kushner, makes this board compatible with any TRS-80 installation.

Circuit Description

Complete address-decoding circuitry in Fig. 1 allows you to decide if you want conventional port I/O to the computer or memory-mapped I/O, a feature Z-80 programmers find useful. Two stable on-board clocks provide you with baud rates ranging from 110 to 2400 baud, which may be overkill, inasmuch as the top end for most phone-line

modem communications is about 300 baud.

You might argue that the baud rate selection doesn't go high enough for a computer-to-computer linkup; but if you think about it for a moment, the better way to go would be to use a parallel port I/O, rather than serial I/O. Eight bits at a time could be sent to your external device rather than waiting for the UART to clock them out one at a time. Parallel is indeed best for speedy I/O, but only if you have an external device capable of going that fast. Try to find a $400 printer that goes 19.2 kilobaud.

Varied Flexibility

Nifty options abound on this board. Besides the usual UART stop bit, data bit and parity selection, you can tie two more status bits up to the board in addition to a DTR (data terminal ready) line. All options are DIP-switch selectable, but not software selectable, which has its good and bad points.

First, fewer parts are needed, so reliability and cost are improved ($79.95, assembled). Second, the amount of driver software is drastically reduced. Consider the size of the short printer-driver routine in Listing 1. A software-selectable I/O scheme would require considerable introductory statements, with perhaps an ongoing lookup routine to remind your board how it is to behave.

On the minus side, you can't input data at one baud rate and output it at another. Nor can you change baud rates without flipping a switch. Big deal. I would never have the occasion to do either. Flipping a DIP switch to change baud rates is not too much bother. How often would you be doing it?

Software Makes It Happen

You can easily use this board to convert your TRS-80 to a stand-alone terminal. I wrote a simple program (Listing 2) to do it all. Just plug in a modem and violã . . . you're a terminal! How about connecting your computer to another via phone lines using modems so you can trade programs or data?

Running the program in Listing 3 will allow you to get bytes of data or whole programs from an external computer. In fact, the external computer can even run programs on your machine with all the control options. To return control back to you, simply tap the BREAK key. Using the extra status bits to detect a ringing pulse on the line, together with an enlarged version of this program, could instantly put you in the micro-time-share business for a mere $79.95. If you are enterprising, this board is more than a conventional good buy . . . it's a dream come true! ■

```
10 CLS: REM ELECTRONIC SYSTEMS 16K TERM PROGRAM
20 PRINT"TERMINAL PROGRAM FOR SERIAL I/O BOARD"
30 PRINT"VERSION 1.0     1979 STEPHEN GIBSON"
40 POKE 16553,255 :PRINT
50 PRINT"THIS PROGRAM POKES A MACHINE LANGUAGE TERMINAL PROGRAM"
60 PRINT"INTO YOUR MEMORY STARTING AT 32000 DECIMAL. THE PROGRAM"
70 PRINT"RUNS DUPLEX AND IGNORES LINE FEEDS. SET YOUR MODEM TO"
80 PRINT"HALF-DUPLEX TO SEE WHAT YOU TYPE ON THE CRT. YOU MUST"
90 PRINT"RUN YOUR MODEM FULL-DUPLEX TO TALK TO TIME SHARE."
100 FOR S = 32000 TO 32048: READ G: POKE S,G :NEXT
110 PRINT: PRINT"ARE YOU RUNNING : "
120 PRINT"1 = ROM BASIC"
130 PRINT"2 = DISK BASIC"
140 INPUT"WHICH";A
150 IF A = 2 THEN 170
160 POKE 16526,00 : POKE 16527,125 :X = USR(0)
170 DEFUSR1=&H7D00:X = USR1(0)
180 DATA 49,0,128,1,232,55,237,120,205,16,125,205
190 DATA 33,125,24,248,10,230,8,200,237,120,230,127
200 DATA 254,10,40,244,205,51,0,24,239,205,43,0
210 DATA 183,200,245,10,230,36,254,36,32,249,241,2,201
```

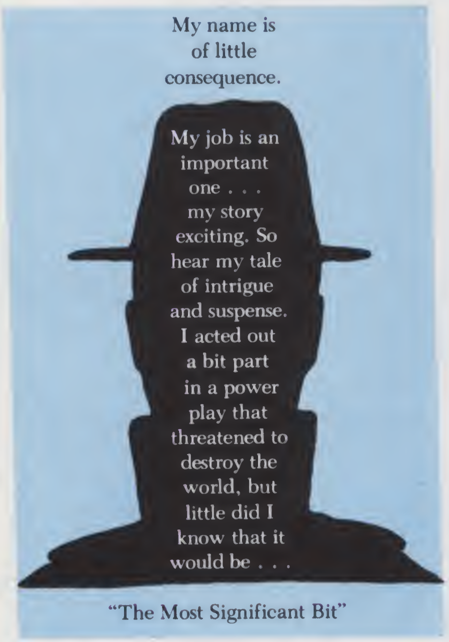
Listing 2. TRS-80 terminal program. Use the TRS-80 as a stand-alone terminal.

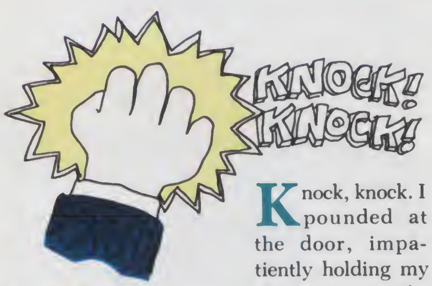
```
10 '  ELECTRONIC SYSTEMS BASIC SERIAL I/O DRIVER
20 CLS:'REV 1.2     1979 STEPHEN GIBSON
30 PRINT"THIS PROGRAM WILL ALLOW YOU TO LOAD A BASIC PROGRAM"
40 PRINT"FROM ANOTHER COMPUTER OR TERMINAL USING THE ELECTRONIC"
50 PRINT"SYSTEMS TRS-80 SERIAL I/O BOARD.":PRINT
60 PRINT"THE PROGRAM IS PLACED AT THE TOP OF YOUR MEMORY. YOU MUST"
70 PRINT"RESERVE SPACE FOR IT WHEN YOU POWER-UP. 'BREAK' FROM THIS"
80 PRINT"PROGRAM, RESET AND ENTER THE APPROPRIATE MEMORY SIZE"
90 PRINT"BELOW FOR YOUR SYSTEM. THEN RELOAD AND CONTINUE THE PROGRAM."
100 PRINT:PRINT" 4K = 20421":PRINT"16K = 32711"
110 PRINT"32K = 49095":PRINT"48K = 65479"
120 PRINT:INPUT"EITHER 'BREAK' OR PRESS 'ENTER' TO CONTINUE.";A$
130 INPUT"NEW MEMORY SIZE";M
140 M = M + 1 : H = INT(M/256) : L = M - 256 * H
150 IF M > 32767 THEN M = M - 65536
160 FOR J = 0 TO 54
170 READ Y
180 POKE M+J,Y
190 NEXT
200 MD=M+16
210 CLS:PRINT"IS THIS PROGRAM RUNNING UNDER...."
220 PRINT"1) DISK BASIC (2.1 DOS)"
230 PRINT"2) ROM BASIC"
240 PRINT:INPUTA:PRINT"PUSH 'BREAK' TO RESTORE CONTROL":IF A=2THEN 290
260 DEFUSR1 = M
280 X=USR1(MD):GOTO 300
290 M=M+16:POKE16526,L:POKE16527,H:X=USR(M)
300 END
310 DATA 205,127,10,34,22,64,197,1
320 DATA 248,55,237,120,193,0,0,201
330 DATA 197,1,248,55,58,127,56,58,64
340 DATA 56,230,4,32,16,10,230,8,40
350 DATA 8,237,120,230,127,254,10,40
360 DATA 233,193,183,201,33,227,3,34
370 DATA 22,64,195,25,26
```

Listing 3. You can input programs or run your TRS-80 from afar with this program. You can also expand it to turn your TRS-80 into a call-up computer just like the big time-share outfits.

Fred LaPointe
119 S. Chestnut St.
Lansdale PA 19446

The Man from C.P.U.





Knock, knock. I pounded at the door, impatiently holding my request for information high, waiting to be acknowledged. The door slowly opened to grant me access. There I was, staring at the stately butler, obviously some lackey peripheral for the rich ol' geezer who bought this fabulous joint from the fortune he made in munitions.

"Yes?"

"I want to see your boss."

"I'm sorry, sir, but the master is occupied at the moment."

So he's trying to give me that "old master's busy" routine, huh. I pulled out my badge to give him an idea of what a top priority I had in his snobbish resolution scheme.

"Very well . . . if you'll walk this way."

I trailed him like a voltage follower into an anteroom and had a seat by the office door. I hated spending wait states in queues, and I considered generating an interrupt by busting the door down. From inside the office came three muffled voices—three angry voices! I heard heated conversation that erupted into angry yelling. I couldn't decode the words but it sounded like someone just blew his stack! Suddenly, the door swung open and out came a woman full of rage—and beauty! Wow! She filled the number six on her football jersey like a hex goddess, and she had a figure like the octal base. What an architecture!

" . . . and I hope I never see you again!" she yelled, flailing her pompon, stomping out of the office. I hardly wished the same of her, I thought, as I sauntered into the rich ol' geezer's office.

"You! Why are you pestering me again?!" From behind the enormous desk the rich ol' geezer looked menacingly at me with his small shift-register eyes.

"Just came to satisfy my curiosity," I said, lighting a cigarette.

"Your curiosity! Why—why—Why you insouciant wastrel!"

The rich ol' geezer was using the pompous high-level language again, but I played his game to give him the idea that he was dealing with low-level logic.

"Did you come here just to raise my blood pressure? Now leave at once!"

"First, tell me what the initials J. K. mean to you," I said, pointing my cigarette like a logic probe.

"J. K. . . . how did you get . . . " The rich ol' geezer's eyes opened wide in amazement, then cold fear. He started going into convulsions and stuttered like a processor doing too much time-sharing.

"They . . . must . . . know . . . "

The rich ol' geezer's emf must have been dropping fast as he clutched the pacemaker at his chest. Slumping to the desk he started writing on a sheet of paper. The writing ended in a serial stream of cryptic scrawls as he crashed with a thud to the desk. Poor guy—there was no way to bring him back on line. Was it a power failure with no backup batteries or had someone deliberately crashed his system? Behind me I heard the door opening.

What do I do now? Run or stay and be pinned with a murder rap? A sticky situation. I incremented my program counter and fetched my next instruction. Of course! I had to find out who the killer was.

So I stuffed the paper into my hat, hoping that it would give me a clue. Then I bounded out the open window into the garden below. Strange! Looked like someone had fallen into this garden before me. Well, I had to go somewhere and do some deep processing. I hopped onto a bus to get to a restaurant on the other side of town.

"Don't anybody move! This is holdup!"

Just my luck. I don't need any distractions and some young punk—a young masked interrupt holding up the bus—waves a heater at the bus driver.

"Better stop, punk. Get out while you're alive!" I yelled from the empty aisle. I caught his attention. "Punk, in this world there are ones and zeros, and you're just a zero!"

"Don't push me,

Pop!" He waved his gun in my direction.

"And don't razz me, punk, or I'll blow you to bits!"

Shots were fired at a synchronous rate, but he was falling; my response time was quicker.

"A man's got to know his limitations," I frowned, blowing the smoke off my magnum.

Several nanoseconds later I was on the other side of town for lunch at "The Menu." As I walked in, the hat girl checked my parity and I flipped here an even two bits. I sat at a table with my back to the wall and pondered the past events. Just as I was about to complete a bite transfer of my kraut dog it hit me like an MOS circuit in a thunderstorm!

"That hat-check girl was the same babe at the poor rich ol' geezer's place. And I left her my hat that had the paper stuffed inside!"

Cursing my bubble memory, I scrambled back to the checkroom. There she sat casting alluring glances . . . a different outfit, but none of those keys were debounced!

"So, you didn't want to leave Meg all by her lonesome," she drawled, putting her arms around my neck to break down my resistance.

"What's that behind your back, babe?"

Instead of supplying the data she tempted me with a lewd interface design, but I pushed her aside retaining my high impedance and morals. This girl wasn't the least bit inhibited!

"Now give me that hat!"

Pow!

O-o-oh! My head. I slowly regained consciousness. I felt like I'd been hit with a line driver. I didn't see that hardware she was packing in her purse. I'd been too gentle with her—another soft error on my part. Where was I? Some dingy basement in a random-seedy part of town.

"Josh, he's awake. You want me hit him again?!"

Hoovering over me was a large-scale thug who seemed to have as much intelligence as an erased EPROM.

"Me crush him like the Indian!" he said picking up a nearby filing cabinet.

Holy semiconductor! This must be the same massive compiler that executed Obj Ekprog Ram, our Indian agent!

"No, Lenny," said a voice which I took to be that of Josh. "Let me ask him a few questions first. Just who are you?"

"I'm Davey Jones, leader of the Monkees."

"Cut the comedy. You're an agent of C.P.U."

"Negative!"

"Then let me refresh your memory!"

The sinister Josh leveled a dagger at my skull, threatening direct memory access. It was now or never! With fury I exploded into Josh's soft gut and desperately jumped to another location as the Compiler tried deleting me with a file. Lenny bounded at me with his huge fists, but I flip-flopped him with a karate move. Now safely grounded, I rammed a downer down his throat that would temporarily turn his mind into an empty scratchpad. I walked over to Josh, who still lay disabled.

"Now, mister, the tables are inverted. You're gonna answer some of my questions!" I said asserting myself. "Just who are you?"

"Josh Klodnicki," he wheezed.

An LED flashed in my head. Of course! This was the infamous J. K. who had been blackmailing the rich ol' geezer for the combination to the vault, which held an illegal micro-atomic bomb built from a hobbyist's magazine. This was the third voice in the office. It was all beginning to add up in the accumulator! He sees the good guys closing in so he rubs out the rich ol' geezer so nobody gets the bomb. Then he jumps out the window for his getaway. Except the program doesn't run the way he writes it. It never does. Before the rich ol' geezer crashes in his chips he writes an important note.

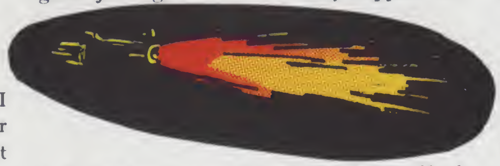
"The hat!" I exclaimed. But lost in thought I had given J. K. a chance to go for his piece. I went for my gun in the twinkling of a seven-segment display and cocked the Schmitt trigger.

"Looks like a stand-off, J. K. Give up and I'll see that you don't get the electric chair."

"Some bargain from a C.P.U. agent. It's either you or me!"

"Very well," I grimaced, "It's a duel, J. K."

Suddenly shots fired, but from neither of our guns! J. K. groaned. I was only clipped, but



J. K. was terminated. I heard the clatter of high heels up the steps and then the slam of a car door. I rushed outside as a sports car peeled away.

Of course! The babe! The woman was double-crossing everybody. She was the rich ol' geezer's mistress as well as J. K.'s blackmailing lady. She wasn't content with a nibble anymore but wanted the whole sector! I had to retrieve that paper. I had to follow that car!

My luck was getting better. I hopped on a 650 Harley-Davidson parked outside and revved 'er up. I hoped the owner wouldn't mind my cycle stealing. I kept her car in sight and kept my distance. Like a phase-locked loop I tracked her to an abandoned bank building. She entered by way of a back door, and I softly single-stepped behind her. She was frantically working on the vault combination lock as I entered.

"Fifty-five," she whispered to herself. Click, click, click. "Fourty-four." Click, click. She grasped the vault lever and pulled, but it wouldn't budge, still latched as tight as a D-flop.

"I've met a lot of characters in my field, but you take the cake, sister!"

She wheeled around in surprise and disgust.

"You fool. We could both make a million!"

I pondered Meg's meg.

"We'll make a deal!" she cried.

She wore a tight miniskirt, and she started raising her address line. She hoped to bend my will like a floppy diskette.

"That won't work, babe!" I said, vowing not to scan her attractive display. "You see, I like my work. It may not pay well, but it's a real character generator. Now give me that paper!"

She motioned the surrender of the paper. Suddenly a deadly purse sailed through the air aimed at my head. This babe's iron was still hot! I fired and blew it away like a random logic design. She froze in fear.

"It's the end of your routine, kiddo. It was quite an operation you were running. You excite the rich ol' geezer beyond his potential. Then you give him weak batteries. You double-cross your boss and wipe him out! Now you've got the paper with the combination on it. Except the program doesn't run the way you write it. It never does!"

"What are you talking about?"

"The paper, sweetheart. The paper doesn't have the combination on it. It's about as useful as write-only memory!"

"But the numbers!"

"Yes, the numbers: 52-4F-53-45-42-55-44. Weren't you ever curious about the 4F?"

"4F? I thought it was a 41."

"No, babe, and you were never gonna get that vault open. You see, the rich ol' geezer had a couple of secrets. And aside from the bomb one of his best kept secrets was that he was an avid microcomputer hobbyist.

"Microcomputer?? What the ----"

"There isn't a combination on that paper. There's only ASCII text."

"ASCII text?! What the hell are you talking about?"

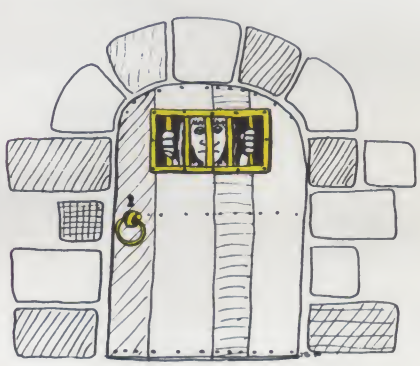
"Translate it in jail you byte-neophyte!"

"You, you . . . "

"Can it in a hermetic package, sweetheart. I'm taking you to the big house."

So that's my story. The bomb was disarmed and taken care of. Now I'm sitting in my dark apartment chugging a beer and watching the tube over a CRT dinner. Once again the world is safe for humanity and some future catastrophe. And you can thank me, the man from C.P.U. Ain't that right, Fifo?

"Woof, woof!" ■



We see the Apple a little differently.

We see it as a good way to get things done.

Apple has built a great computer. We at CCS have built a great line of peripherals and components to expand the Apple. To do almost anything you want to get done with a computer.

If you want to do business with an Apple, we've got tools to connect the Apple to standard business printers and terminals. Or to modems, for communications over telephone lines, with other computers, even with other Apples.

If you want to apply your Apple to engineering, scientific, or graphic projects, we've got tools for high-powered, high-speed math functions, and fast, high resolution graphics. And tools to connect the Apple to lab test equipment like function generators or plotters.

And we have tools to connect the Apple to the outside world, including A/D converters and interval timers with external interface.

We make components for the S-100 bus, the PET, and the TRS-80, too. We built our products to deliver hard-nosed value to the OEM, and to the inventor who knows the best, at prices that are unbeaten.

To find out how much computer your Apple II can be, see things our way. Because for serious users with serious uses for the Apple, we've got the tools.

California Computer Systems
250 Caribbean Sunnyvale, CA 94086 (408) 734-5811

Meet the Paper Tiger

Evaluation of IDS's impact printer finds it lightweight, surefooted and easy to care for.

George H. Brooks
Industrial Engineering
Auburn University
Auburn AL 36830

I recently saw a brief demonstration of Integral Data Systems' (14 Tech Circle, Natick MA 01760) IDS-440 impact printer, called the Paper Tiger. I was impressed enough to order one for evaluation and use in our Industrial Engineering Microcomputer Laboratory at Auburn University. Having now used it in a number of configurations, I have been pleased by its performance and ease of use, and feel that it may rank in the "best buy" category.

Specifications

One first notices the small size of the unit and its light weight—approximately 12.5 inches (32 cm) high, 15.75 inches (40 cm) wide and 12.5 inches (32 cm) deep and 20 pounds (7.5 kg). Its dimensions make the unit readily movable, although in the lab it normally sits on top of our North Star Horizon, with which we use it most frequently.

Pinfeed paper ranging in width from 1.75 inches to the stock width of 9.5 inches must be used. Forms length may vary from 3 to 14 inches, with a number of intermediate sizes selectable by DIP switches. Fanfold paper fits under and to the rear of the printer, taking up little space. A roller to use roll paper is available as an option.

The printer uses a 3870 microprocessor and ROM for control. This has resulted in a single board configuration that also includes 256 bytes of input buffer RAM in the standard model, or 2K bytes as an option. The option also includes dot graphics capability.

Other key specifications include:

Full uppercase and lowercase printable ASCII characters, plus 13 control codes, one of which pertains only to the graphics option.

Variable character density, 8.3, 10, 12 and 16.5 characters per inch, plus double width (enhanced) characters in each density. Character density can be set either by DIP switches or under program control. Enhanced mode characters are invoked only under program control. Printer select and deselect and graphics mode (if so equipped) are program controlled.

Serial EIA RS-232C or parallel TTL-level interface. This latter interface (which we have not used) is Centronics compatible.

Print speeds are variable and depend on character density, among other factors. Maximum speed is 198 characters per second; highest sustained speeds range from 45 cps at 8.3 cpi to 92 cps at 16.5 cpi.

All operator controls are readily accessible. The main power switch is on the rear panel, but since the unit is small, it is easy to reach. A line fuse and a 115/230 volt select switch, which is recessed to prevent accidental use, are also located on the rear panel. All other controls are at the top left and top right of the unit.

At the top left is the formset/test switch, which is used to print a built-in test pattern and to set top of form. Also at the top left are two banks of DIP switches used for the less frequently changed settings such as baud rate, form length and print density.

At the top right of the printer are two more operator switches: the offline/online switch and the formfeed/linefeed switch. Indicator lights for power on, on line and paper out are also located in this control cluster.

Owner's Manual

We received our unit, via UPS from the manufacturer, late in the afternoon. We opened the package, following the directions on the outside of the package, and encountered the owner's manual and perhaps 50 sheets of standard paper.

Next we uncovered the printer, sitting on a heavy cardboard square, and enclosed in a heavy film wrapper. Since the hour was late, we merely removed the printer from the packing, resisting the urge to operate it. I took the manual home for perusal during the evening.

kilobaud
MICROCOMPUTING™

The Integral Data Systems, Inc., IDS-440 Paper Tiger. Note the operator controls at the top right. Other controls are at the left, hidden in the photo by the paper. (Photo courtesy of Integral Data Systems)

The manual is outstanding. Comprising six major sections and two appendices, it is copiously illustrated. The first section covers the characteristics and specifications. The second section, detailing the installation and configuration of the unit, is well done with clear text, pictures and figures illustrating connections, switch settings, timing and other installation considerations.

The third section deals with operator controls and indicators; while the fourth section contains an informative and detailed description of the principles of operation. Section 6 reviews the graphics option and the internal paper-roll-holder option.

The fifth section ("Maintenance and Troubleshooting") is worthy of particular note, especially for persons who do their own maintenance. It seems to encourage the owner to "do it yourself," particularly considering the inclusion of Appendix A, which contains complete schematics of the power supply and main logic board. Why can't every manufacturer include schematics?

Appendix B contains detailed instructions for preparation of a Centronics-compatible cable, which should enable you to interface the printer to a host of computers, including the ubiquitous TRS-80.

The maintenance section contains detailed procedures for many maintenance functions that are required of all such equipment, but are frequently not even mentioned in other manufacturers' manuals. For example, platen adjustment, paper drive belt tension, printhead carriage lubrication and printhead drive belt tension are covered in detail, along with more mundane matters such as paper loading and ribbon replacement. A detailed section is also included on printhead cleaning and lubrication, subjects which are studiously avoided in most other user manuals.

In addition, a three-page table of troubleshooting hints details possible causes of problems. One of these hints led us quickly to a problem in another piece of equipment which had initially caused us difficulty in printing at high baud rates.

Installation and Use

Armed with the manual, I arrived early at our laboratory the next morning. My first trial was trivial, so far as difficulty was concerned. Following the directions of section 2 of the manual, I completed the unpacking, loaded some paper and ran a test pattern using the format/test and offline/online switches.

Having no problems, I then cabled the printer to an Infoton 100 terminal, which is the principle terminal for our Horizon. The Infoton has a Function-Print command—available from the keyboard—which outputs whatever is on the screen through an auxiliary RS-232-type port to a printer.

I set the Paper Tiger, the Infoton and the Horizon to 1200 baud, brought up the computer and filled the screen with disk directories. Then came the moment of truth. I depressed Function-Print on the Infoton, and—behold—the Paper Tiger faithfully reproduced every character on the terminal screen.

Emboldened by this success, I then put the Infoton in Copy mode, wherein everything received by the terminal or originated from the keyboard is automatically printed. Tentatively, I typed a DOS command, which was echoed by the printer. The command executed, and the material that came to the screen was again faithfully reproduced by the printer. Certainly, this was immediate and gratifying success. (We encountered some minor problems when printing at 1200 baud after the 2K buffer became full, but quickly discovered that the terminal did not recognize the DTR signal from the printer. We promptly corrected this problem, which does not pertain either to printer design or performance.)

After establishing that the Paper Tiger could function in a copy mode with a CRT-type terminal, we tried it in this same copy mode with a Hazeltine 1500 terminal, which connects through an acoustical coupler to any of our three time-sharing computers on campus.

Again, we encountered *no difficulty*, although the Hazeltine 1500 does not have the capability of being able to copy what is already on the CRT screen. We found, however, that judicious use of the main power

switch on the Tiger could give us the capability of selective copy with little difficulty.

We next turned our attention to using the printer as a direct printer, using a serial output port on the North Star Horizon. Again, we had immediate success. We connected to port 1, brought up the computer and printed at 1200 baud without problem or error.

Finally, we tested the graphics capability. This took a bit more work—not in the interface, but in learning the format and writing and assembling a program to use the graphics mode.

In the graphics mode, the printer will respond to twelve different control sequences. Each sequence consists of a pair of ASCII control characters. The first of each pair is ETX, which commands the graphics mode, while the second character in the pair commands some function, e.g., linefeed (LF), vertical tab (VT), etc.

In graphics mode, the printer interprets any data byte in a strictly binary fashion, which, in turn, causes the seven-vertical-needle printhead to output a column of from zero to six dots. Bits 0 to 5 of each byte are used. Bit six is also read and printed but is overwritten by the next print line, so bit six should always be zero. Bit 7 is ignored.

The printer uses a raster-scan-type of action, and two scans are equivalent to one line of normal character printing. Mixed graphics and character printing are both possible and relatively easy. The logo examples represent the printer's graphics output capability.

The graphics-mode vertical-dot density is approximately 72 dots per inch (0.014 inch between print needle centers). As in character printing, horizontal dot density is a function of selected print density. The nearest approach to equal horizontal and vertical density exists at a 12 cpi print density, where the horizontal distance between print needle centers is 0.0156 inch. This is the setting used in the logo examples.

At this density, an 8 inch print line consists of 513 dots. In an 8 by 10 inch print area, you can achieve good resolution with 369,360 individual dots printed. However, if you used a byte of memory for each 6-dot column of such an output, 61,560 bytes of data would be required.

Fortunately, in practice such memory size is not required; the examples require less than 4K bytes for both data storage and program. This compression occurs because subroutines for standard shapes can be written, then combined repetitively to produce the desired output. For example, a full line of any vertical dot arrangement requires no data storage, and only 19 bytes of instructions. This line can be called repeatedly in a given program.

Observations

The overall quality of the unit, its ease of use and its ease of maintenance are impressive characteristics. The print ribbon deserves special note. The ribbon is contained on typewriter-sized spools, which are readily accessible for changing. The ribbon runs over two inking rollers and is driven by its own ribbon drive motor. Ribbon life is 5 to 10 million characters.

When it is necessary to change the ribbon, then the inking rollers should also be changed. Ribbons and rollers are available from IDS as a set for $12. Since the printhead has an estimated life expectancy of 100 million characters, the ribbon cost is nominal over the life of the printer.

The world is not perfect, and the Paper Tiger has some annoyances. The printer output, left to its own devices, sometimes gets trapped in the inbound paper, causing a minor jam, and usually resulting in overprinted lines. I have "jury-rigged" a wire bail to carry the paper out behind the machine; this seems to work well. An accessory paper tray to catch the output is available for $12.

In the graphics mode, row-to-row registration seems to be sensitive to minor maladjustments of printhead and paper drive belt tensions, and to the positioning of the paper supply pile. With careful adjustment, long vertical or slanted lines appear straight, and row-to-row distance remains relatively constant. Twenty pound paper works better in graphics mode than does sixteen pound, as it is heavier and more stable. In normal print mode, minor maladjustments do not affect the print appearance nearly as much.

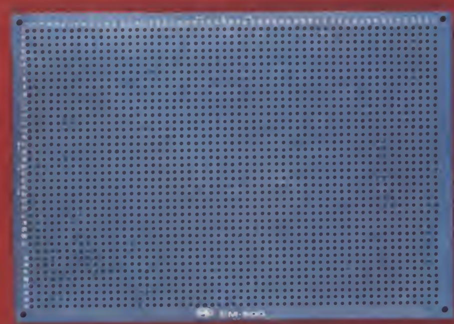
Like most matrix printers, noise during printing is evident. It is difficult for me to appraise this in definite terms, as other equipment in the lab prints at a slower rate. Furthermore, my little lab is typical of university labs—no drapes, no carpet and lots of surfaces for sound to reflect from. When we can, we plan to make some relative sound measurements under controlled conditions.

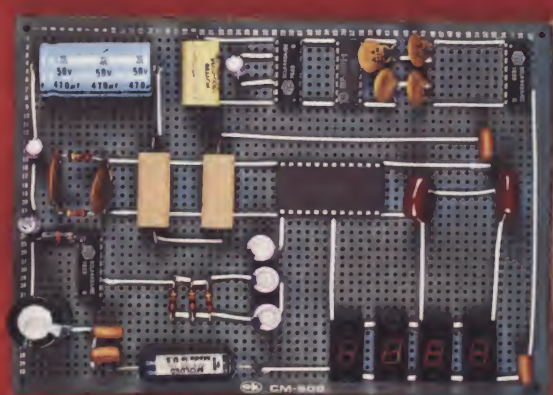
Conclusions

Based on my experience with this unit, I consider the Paper Tiger an outstanding printer, especially in view of its $995 base price, and $199 for the graphics capability and extra buffer. The manual is exceptional in its clarity and inclusion of troubleshooting and maintenance material, including circuit diagrams.

It has three main modes of operation—as a CRT duplicator, as a conventional printer and as a dot graphics printer. With the appropriate CRT, the Paper Tiger can also copy ASCII material as composed on the CRT screen. ■

CM-600 Circuit Mount





CM-600 $6.95*
RW-50 $2.98*

NEW CM-600 SOLDERLESS PROTOTYPE BOARD

CM-600 is a unique system for solderless construction of circuit prototypes, useful to both engineers and hobbyists. The CM-600 is a neoprene board 4½″ (114mm) x 6″ (152mm) with 2280 holes on .100″ (2.54mm) centers. Standard components including DIP's are mounted by simply inserting leads into the holes in the long life neoprene material. Interconnections are easily made using 20 or 22 AWG (0,8 or 0,65mm) wire jumpers. Positive contact is assured by the elasticity of the hole, which compresses the leads together. To remove components or leads, simply pull out. This facilitates easy circuit changes making it ideal for breadboarding experimental circuits. CM-600 also features numbered rows and columns for easy reference.

Accessory Kit RW-50 contains 50 pcs of AWG 20 (0,8mm) insulated jumper wires of assorted lengths from ½″ (13mm) to 4″ (100mm). Both ends are stripped and bent 90° for easy insertion. In stock directly from

OK Machine & Tool Corporation ✓21

3455 Conner St., Bronx, N.Y. 10475 U.S.A.

Tel. (212) 994-6600 Telex 125091

*Minimum billings $25.00, add shipping charge $2.00
New York State residents add applicable tax

Larry Dolce
2 Mayfield St.
Valley Cottage NY 10989

RCA's VIP Tiny BASIC

Cosmac users who have been looking to fill a software void need look no further.

Those of us who purchased RCA's Cosmac VIP microprocessor and have outgrown playing games, but have found little other software, can relax. For $39 RCA offers VIP Tiny BASIC in 4K ROM. The basic board plugs directly into the expansion connector and starts up as soon as you switch the VIP toggle switch to RUN. The only other requirement for running VIP Tiny BASIC is an ASCII keyboard.

Commands

Table 1 contains the VIP Tiny BASIC commands and their respective abbreviations. Some of the abbreviations fail to save keystrokes, so I did not see any real use for these abbreviations. The question mark abbreviation for the print statement is a real time-saver considering the number of times the print statement is used in a program. Another time-saver allowed by VIP BASIC is the omission of the keyword LET when assigning values to variables.

When the program is displayed using the list command, VIP Tiny BASIC replaces all abbreviations with the full spelling.

As in other BASICs, lines can be inserted or deleted. To delete a line, type the line number and return. Inserting a line requires that a line number be available at the place in the program where you wish to insert the new statement. For this reason, it is a good idea to increment line numbers by at least 10 or 20 to allow room for insertion.

Table 2 lists the error messages that VIP BASIC displays. The "What?" error message is convenient because VIP BASIC displays a question mark (?) before the first occurrence of a syntax error (see Photo 1). This makes it easy to identify and correct errors in syntax. The "How?" error will also display a question mark at the point in the statement where the error exists. The remaining error messages are self-explanatory.

Operations and Variables

VIP BASIC contains four arithmetic operations: addition (+), subtraction (−), multiplication (*) and division (/). Conditions are tested using the following relational operations: greater than (>), equal to (=), greater than or equal to (>=), less than (<), not equal to (<>) and less than or equal to (<=).

VIP BASIC variables are A through Z, allowing for one subscripted variable, that is, A(X). The subscripted variable A(X) is not the same as variable A, and both can be used in the same program.

The numeric range for VIP BASIC is the same for many other Tiny BASICs. Since these interpreters perform 16-bit (two-byte) integer arithmetic, the maximum positive value of an integer is 32767. Any value over 32767 becomes a negative number. This allows the range of any number to be between −32768 and +32767.

Use

The first thing I had to become accustomed to was VIP BASIC's speed. My experience with BASIC has been with a national time-sharing service using large mainframe computers. Attempting to program my VIP using BASIC caught me by surprise. This BASIC is not very fast, especially when it comes to calculations.

One way to increase processing speed is to use the TVOFF and TVON commands. Turning the display off and then back on will allow the VIP to process in about 50 percent of the time required with the display on.

The reason for the increase in speed concerns the video interface. The 1861 video IC interrupts the 1802 microprocessor 60 times per second to refresh

Command	Abbreviation	Operation
New	N.	Clears program storage area
List	L.	Displays program starting at lowest line number
List n		Displays program starting at line n
Run	R.	Executes program beginning at lowest line
Go To n	G.	Branch to line number n
Go Sub n	GOS.	Call subroutine at line n
Return	RET.	Return from subroutine
If (exp.) then n	I. T.	Tests expression and, if true, branches to line n
Input	IN.	Input numeric data only
Let	LE.	Assigns value to a variable
Print	P. or ?	Prints information on screen
Print at X,Y	PA. or ?A	Prints information on screen at coordinates of X and Y
Rem	R E.	Allows remarks in program
ABS (X)	A.	Absolute value of expression X
RND	R.	Returns a random number from 0 to 255
END	E.	Halts program execution
SAVE	S.	Stores program on cassette
LOAD	LO.	Loads program from cassette
CLS	CL.	Clears screen
COLOR	C.	Sets color (requires color board)
Go Key	GOK.	Branch on any key pressed
KEY	K.	Contains ASCII value of key pressed in Go Key statement
TI		Sets internal timer
FQ n	F.	Sets tone frequency (requires simple sound board) to value of n
TO n		Sets tone duration of value of n
Show X, Y	SH.	Display pattern of variable PT at coordinates of X and Y
PT n		Special variable to display bit pattern of n with show command
TV On		Turns screen on
TV Off		Turns screen off
HIT	H.	Special variable that determines if a hit occurred to a pattern in the last show or print at statement
MEM	M.	Displays program storage space remaining

Table 1. VIP Tiny BASIC commands.

Error Message	Cause
What?	Syntax error
How?	Not enough information
LD ERR	Tape read error
ERR	Invalid data for input command
Sorry	No more memory

Table 2. VIP Tiny BASIC error messages.

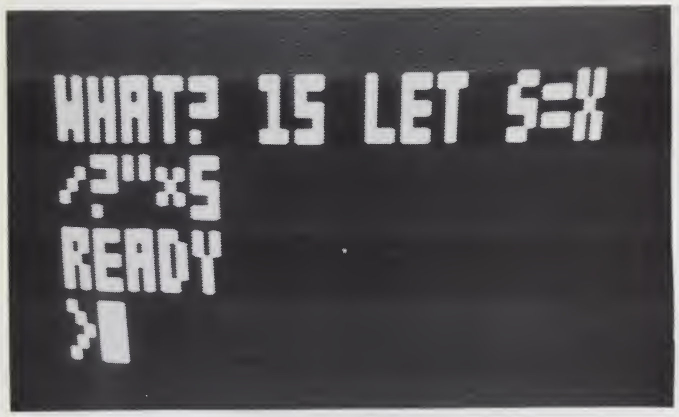


Photo 1. The "What?" error message. Note position of "?" before quotes. This indicates where the syntax error occurred.

the display. Turning off the video allows the 1802 to process without interruption.

Another aspect of VIP BASIC that required a little acclimation is the display itself. The RCA VIP BASIC uses a 64 by 32 bit display page. Utilizing this display map, you can display five lines of 16 characters if each character occupies a 4 by 6 bit matrix. Each character must fit in an area six bits high by four bits wide. This is not much room to display letters such as W and M. (See Photo 2 for an example of the VIP basic character set. All characters are shown except for ↑, which is not decoded on my GRI keyboard.)

After becoming accustomed to the display, I began structuring my programs using 16 characters or less per line. Having the ability to use Tiny BASIC on my VIP gave me the feel of programming, which far outweighs the shortcomings of this BASIC.

The Manual

The manual supplied with the BASIC ROM board is directed toward the beginner. An explanation of the BASIC programming language is given, along with an attempt to provide a feeling for programming.

Having been a professional programmer, I have found no problems in understanding the RCA manual. I would recommend to anyone interested in programming VIP BASIC or, for that matter, any other programming language to read as much as possible on the use of the language.

Conclusion

I like VIP Tiny BASIC. There is enough substance in this BASIC to keep even a "professional" programmer happy. If you already own an RCA Cosmac VIP microprocessor, you should consider acquiring VIP Tiny BASIC. ■

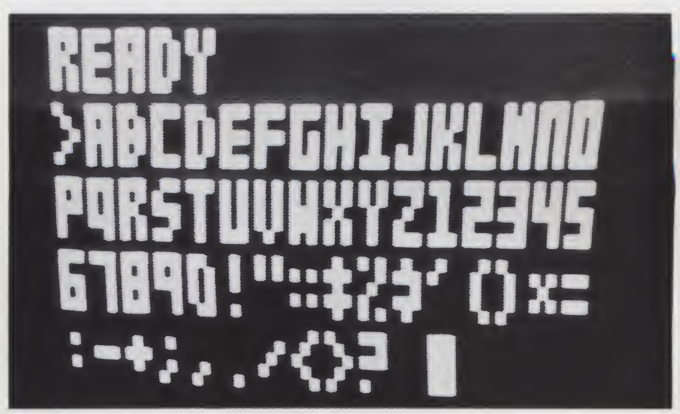


Photo 2. VIP BASIC character set.

DIGITAL RESEARCH

Software with Manual / Manual Alone

- ☐ **CP/M FLOPPY DISKETTE OPERATING SYSTEM** — Ⓜ Packages supplied on diskette complete with 8080 assembler, text editor, 8080 debugger and various utilities plus full documentation. CP/M available configured for most popular computer/disk systems including: North Star Single, Double or Quad density, Altair 8" disks, Helios II, Exidy Sorcerer, Vector MZ, Heath H17† or H89†, TRS-80†, iCOM 3712 and iCOM Micro Disk plus many other configurations available off the shelf ..... $145/$25
  CP/M version 2 (not all formats available immediately) ..... $170/$25
- ☐ **MP/M*** ..... $300/$50

Lower prices!

- ☐ **MAC** — 8080 Macro Assembler. Full Intel macro definitions. Pseudo Ops include RPC, IRP, REPT, TITLE, PAGE, and MACLIB. Z80 library included. Produces Intel absolute hex output plus symbols file for use by SID (see below) ..... $85/$15
- ☐ **SID** — 8080 symbolic debugger. Full trace, pass count and break-point program testing system with back-trace and histogram utilities. When used with MAC, provides full symbolic display of memory labels and equated values ..... $70/$15
- ☐ **ZSID** — As above for Z80. Requires Z80 CPU $95/$25

NEW!!

- ☐ **TEX** — Text formatter to create paginated, page-numbered and justified copy from source text files, directable to disk or printer ..... $70/$15
- ☐ **DESPOOL** — Program to permit simultaneous printing of data from disk while user executes another program from the console ..... $45/$5

All Microsoft prices are discounted!

MICROSOFT

- ☐ **BASIC-80** — Disk Extended BASIC, ANSI compatible Ⓛ with long variable names, WHILE/WEND, chaining, Ⓜ variable length file records ..... $300/$25
- ☐ **BASIC COMPILER** — Language compatible with Ⓛ BASIC-80 and 3-10 times faster execution. Produces Ⓜ standard Microsoft relocatable binary output. Includes Macro-80. Also linkable to FORTRAN-80 or COBOL-80 code modules ..... $350/$25
- ☐ **FORTRAN-80** — ANSI 66 (except for COMPLEX) plus Ⓛ many extensions. Includes relocatable object compiler, linking loader, library with manager. Also in-Ⓜ cludes MACRO-80 (see below) ..... $400/$25
- ☐ **COBOL-80** — Level 1 ANSI '74 standard COBOL plus Ⓛ most of Level 2. Full sequential, relative, and in-Ⓜ dexed file support with variable file names. STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 ..... $625/$25
- ☐ **MACRO-80** — 8080/Z80 Macro Asembler. Intel and Ⓛ Zilog mnemonics supported. Relocatable linkable Ⓜ output. Loader, Library Manager and Cross Reference List utilities included ..... $149/$15
- ☐ **XMACRO-86** — 8086 cross assembler. All Macro and Ⓛ utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available ..... $275/$25
- ☐ **EDIT-80** — Very fast random access text editor for text Ⓛ with or without line numbers. Global and intra-line commands supported. File compare utility included ..... $89/$15

MICRO FOCUS

- ☐ **STANDARD CIS COBOL** — ANSI '74 COBOL stand-Ⓛ ard compiler fully validated by U.S. Navy tests to ANSI level 1. Supports many features to level 2 including dynamic loading of COBOL modules and a full ISAM file facility. Also, program segmentation, interactive debug and powerful interactive extensions to support protected and unprotected CRT screen formatting from COBOL programs used with any dumb terminal ..... $850/$50
- ☐ **FORMS 2** — CRT screen editor. Output is COBOL data Ⓛ descriptions for copying into CIS COBOL programs. Automatically creates a query and update program of indexed files using CRT protected and unprotected screen formats. No programming experience needed. Output program directly compiled by CIS COBOL (standard) ..... $200/$20

- ☐ *HDBS* — Hierarchical Data Base System. CODASYL oriented with FILEs, SETs, RECORDs and ITEMs which are all user defined. ADD, DELETE, UPDATE, SEARCH, and TRAVERSE commands supported. SET ordering is sorted, FIFO, LIFO, next or prior. One to many set relationship supported. Read/Write protection at the FILE level. Supports FILEs which extend over multiple floppy or hard disk devices.
- ☐ **MDBS** — Micro Data Base System. Full network data base with all features of HDBS plus multi-level Read/Write protection for FILE, SET, RECORD and ITEM. Explicit representation of one to one, one to many, many to many, and many to one SET relationships. Supports multiple owner and multiple record types within SETs. HDBS files are fully compatible.
- ☐ **MDBS-DRS** — MDBS with Dynamic Restructuring System option which allows altering MDBS data bases when new ITEMs, RECORDs, or SETs are needed without changing existing data.

NOTE

HDBS-Z80 version ..... $250/$35
MDBS-Z80 version ..... $750/$35
MDBS-DRS-Z80 version ..... $850/$35
8080 Version available at $75. extra.

Z80 version requires 20K RAM. 8080 version requires 24K RAM. (Memory requirements are additional to CP/M and application program.)

When ordering HDBS or MDBS please specify if the version required is for 1) Microsoft L80 i.e. FORTRAN-80, COBOL-80, BASIC COMPILER, 2) MBASIC 4. XX, or 3) BASIC-80 5.0.

Prices and specifications subject to change without notice.

EIDOS SYSTEMS

Lower prices!

- ☐ **KISS** — Keyed Index Sequential Search. Offers com-Ⓛ plete Multi-Keyed Index Sequential and Direct Access file management. Includes built-in utility functions for 16 or 32 bit arithmetic, string/integer conversion and string compare. Delivered as a relocatable linkable module in Microsoft format for use with FORTRAN-80 or COBOL-80, etc. ..... $335/$23
- ☐ **KBASIC** — Microsoft Disk Extended BASIC with all Ⓛ KISS facilities, integrated by implementation of nine additional commands in language. Package includes KISS. REL as described above, and a sample mail list program ..... $585/$45
  To licensed users of Microsoft BASIC-80 (MBASIC) ..... $435/$45

All Micropro prices are discounted!

MICROPRO

- ☐ **SUPER-SORT I** — Sort, merge, extract utility as abso-Ⓛ lute executable program or linkable module in Micro-Ⓜ soft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating, fixed point, exponential, field justified, etc. Even variable number of fields per record! ..... $225/$25
- ☐ **SUPER-SORT II** — Above available as absolute pro-Ⓛ Ⓜ gram only ..... $175/$25
- ☐ **SUPER-SORT III** — As II without SELECT/EXCLUDE Ⓛ Ⓜ ..... $125/$25
- ☐ **WORD-STAR** — Menu driven visual word processing Ⓛ system for use with standard terminals. Text format-Ⓜ ting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning ..... $445/$40
- ☐ **WORD-STAR/MAIL-MERGE** — As above with option Ⓛ for production mailing of personalized documents Ⓜ with mail list from Datastar or NAD ..... $575/$25
- ☐ **WORD-STAR Customization Notes** — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR ..... NA/$100
- ☐ **WORD-MASTER** Text Editor — In one mode has super-Ⓛ set of CP/M's ED commands including global search-Ⓜ ing and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal ..... $125/$25
- ☐ **DATASTAR** — Professional forms control entry and Ⓛ display system for key-to-disk data capture. Menu Ⓜ driven with built-in learning aids. Input field verification by length, mask, attribute (i.e. uppercase, lowercase, numeric, auto dup., etc.). Built-in arithmetic capabilities using keyed data, constants and derived values. Visual feedback for ease of forms design. Files compatible with all CP/M-MP/M supported languages. Requires 32K CP/M ..... $350/$35

- ☐ **CBASIC-2** Disk Extended BASIC — Non-interactive Ⓜ BASIC with pseudo-code compiler and runtime interpreter. Supports full file control, chaining, integer and extended precision variables, etc. ..... $109/$15
- ☐ **PASCAL/M** — Compiler generates P code from ex-Ⓘ tended language, implementation of standard PAS-Ⓜ CAL. Supports overlay structure through additional procedure calls and the SEGMENT procedure type. Provides convenient string handling capability with the added variable type STRING. Untyped files allow memory image I/O. Requires 56K CP/M ..... $350/$30
- ☐ **PASCAL/Z** — Z80 native code PASCAL compiler. Pro-Ⓘ duces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, companion macro-assembler and source for the library. Requires 56K and Z80 CPU.
  Version 2 includes all of Jensen/Wirth except variant records ..... $275/$25
  Version 3 Upgrade with variant records and strings expected 3/80 ..... $395/$25
- ☐ **PASCAL/MT** — Subset of standard PASCAL. Gener-Ⓧ ates ROMable 8080 machine code. Symbolic debug-Ⓜ ger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Version 3 includes Sets, Enumeration and Record data types. Manual explains BASIC to PASCAL conversion. Source for the run time package requires MAC (See under Digital Research). Requires 32K ..... $250/$30

New Version
New Features

STRUCTURED SYSTEMS GROUP

- ☐ **GENERAL LEDGER** — Interactive and flexible system †† providing proof and report outputs. Customization of COA created interactively. Multiple branch accounting centers. Extensive checking performed at data entry for proof, COA correctness, etc. Journal entries may be batched prior to posting. Closing procedure automatically backs up input files. Now includes Statement of Changes in Financial Position. Requires CBASIC-2 ..... $1250/$25
- ☐ **ACCOUNTS RECEIVABLE** — Open item system with †† output for internal aged reports and customer-oriented statement and billing purposes. On-Line Enquiry permits Information for Customer Service and Credit departments. Interface to General Ledger provided if both systems used. Requires CBASIC-2 ..... $1250/$25
- ☐ **ACCOUNTS PAYABLE** — Provides aged statements †† of accounts by vendor with check writing for selected invoices. Can be used alone or with General Ledger and/or with NAD. Requires CBASIC-2 ..... $1250/$25
- ☐ **PAYROLL** — Flexible payroll system handles weekly, †† bi-weekly, semi-monthly and monthly payroll periods. Tips, bonuses, re-imbursements, advances, sick pay, vacation pay, and compensation time are all part of the payroll records. Prints government required periodic reports and will post to multiple SSG general ledger accounts. Requires CBASIC-2 ..... $1250/$25
- ☐ **INVENTORY CONTROL SYSTEM** — Performs control †† functions of adding and depleting stock items, adding new items and deleting old items. Tracks quantity of items on hand, on order and back-ordered. Optional hard copy audit trail is available. Reports include Master Item List, Stock Activity, Stock Valuation and Re-order List. Requires CBASIC-2 $1250/$25

- ☐ **ANALYST** — Customized data entry and reporting sys-†† tem. User specifies up to 75 data items per record. Interactive data entry, retrieval, and update facility makes information management easy. Sophisticated report generator provides customized reports using selected records with multiple level break-points for summarization. Requires CBASIC-2 ..... $250/$15
- ☐ **LETTERIGHT** — Program to create, edit and type letters or other documents. Has facilities to enter, display, delete and move text, with good video screen presentation. Designed to integrate with NAD for form letter mailings. Requires CBASIC-2 ..... $200/$25
- ☐ **NAD** Name and Address selection system — interactive mail list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. Requires CBASIC-2 ..... $100/$20
- ☐ **QSORT** — Fast sort/merge program for files with fixed record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created ..... $100/$20

GRAHAM-DORIAN SOFTWARE SYSTEMS

- ☐ **GENERAL LEDGER** — An on-line system; no batch-Ⓛ ing is required. Entries to other GRAHAM-DORIAN Ⓜ accounting packages are automatically posted. User †† establishes customized C.O.A. Provides transaction register, record of journal entries, trial balances and monthly closings. Keeps 14 month history and provides comparison of current year with previous year. Requires CBASIC-2. Supplied in source ..... $995/$35

New Features

- ☐ **ACCOUNTS PAYABLE** — Maintains vendor list and Ⓛ check register. Performs cash flow analysis. Flexible Ⓜ — writes checks to specific vendor for certain in-†† voices or can make partial payments. Automatically posts to GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source ..... $995/$35
- ☐ **ACCOUNTS RECEIVABLE** — Creates trial balance re-Ⓛ ports, prepares statements, ages accounts and rec-Ⓜ ords invoices. Provides complete information describ-†† ing customer payment activity. Receipts can be posted to different ledger accounts. Entries automatically update GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source ..... $995/$35
- ☐ **PAYROLL SYSTEM** — Maintains employee master file. Ⓛ Computes payroll withholding for FICA, Federal and Ⓜ State taxes. Prints payroll register, checks, quarterly †† reports and W-2 forms. Can generate ad hoc reports and employee form letters with mail labels. Requires CBASIC-2. Supplied in source ..... $590/$35
- ☐ **INVENTORY SYSTEM** — Captures stock levels, costs, Ⓛ sources, sales, ages, turnover, markup, etc. Trans-Ⓜ action information may be entered for reporting by †† salesman, type of sale, date of sale, etc. Reports available both for accounting and decision making. Requires CBASIC-2. Supplied in source ..... $590/$35
- ☐ **JOB COSTING** — Designed for general contractors. Ⓛ To be used interactively with other GRAHAM-DORIAN Ⓜ accounting packages for tracking and analysing ex-†† penses. User establishes customized cost categories and job phases. Permits comparison of actual versus estimated costs. Automatically updates GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source $995/$35
- ☐ **APARTMENT MANAGEMENT SYSTEM** — Financial Ⓛ management system for receipts and security de-Ⓜ posits of apartment projects. Captures data on va-†† cancies, revenues, etc. for annual trend analysis. Daily report shows late rents, vacancy notices, vacancies, income lost through vacancies, etc. Requires CBASIC-2. Supplied in source ..... $590/$35
- ☐ **CASH REGISTER** — Maintains files on daily sales. Ⓛ Files data by sales person and item. Tracks sales, Ⓜ over-rings, refunds, payouts and total net deposits. †† Requires CBASIC-2. Supplied in source ..... $590/$35

- ☐ **tiny C** — Interactive interpretive system for teaching Ⓧ structured programming techniques. Manual includes full source listings ..... $75/$40
- ☐ **BDS C COMPILER** — Supports most major features of Ⓜ language including Structures, Arrays, Pointers, recursive function evaluation, linking loader and library. Floating point function library included. Lacks data initialization and static and register class specifiers. Documentation includes "THE C PROGRAMMING LANGUAGE" by Kernighan and Ritchie. ..... $125/$15

New Features

- ☐ **WHITESMITHS C COMPILER** — The ultimate in sys-Ⓛ tems software tools. Produces faster code than Pas-Ⓧ cal with more extensive facilities. Conforms to the full UNIX*** Version 7 C language, described by Kernighan and Ritchie, and makes available over 75 functions for performing I/O, string manipulation and storage allocation. Linkable to Microsoft REL files. Requires 60K CP/M ..... $630/$30
- ☐ **POLYVUE/80** — Full screen editor for any CRT with Ⓜ XY cursor positioning. Includes vertical and horizontal scrolling, interactive search and replace, automatic text wrap around for word processing, operations for manipulating blocks of text, and comprehensive 70 page manual ..... $135/$15
- ☐ **POLYTEXT/80** — Text formatter for word processing Ⓜ applications. Justifies and paginates source text files. Will generate form letters with custom fields and conditional processing. Support for Daisy Wheel printers includes variable pitch justification and motion optimization ..... $85/$15
- ☐ **ALGOL-80** — Powerful block-structured language compiler featuring economical run time dynamic allocation of memory. Very compact (24K total RAM) system implementing almost all Algol 60 report features plus many powerful extensions including string handling direct disk address I/O etc. Requires Z80 CPU ..... $199/$20
- ☐ **Z80 DEVELOPMENT PACKAGE** — Consists of: (1) disk Ⓜ file line editor, with global inter and intra-line facilities; (2) Z80 relocating assembler, Zilog/Mostek mnemonics, conditional assembly and cross reference table capabilities; (3) linking loader producing absolute Intel hex disk file ..... $95/$20

*CP/M is a trademark of Digital Research.
**Z80 is a trademark of Zilog, Inc.
***UNIX is a trademark of Bell Laboratories.
****WHATSIT? is a trademark of Computer Headware.
*****Electric Pencil is a trademark of Michael Shrayer Software.
★TRS-80 is a trademark of Tandy Corp.

†CP/M for Heath, TRS-80 Model I and PolyMorphic 8813 are modified and must use specially compiled versions of system and applications software.

††Recommended system configuration consists of 48K CP/M, 2 full size disk drives, 24 x 80 CRT and 132 column printer.

Ⓜ Modified version available for use with CP/M as implemented on Heath and TRS-80 Model I computers.

Ⓛ User license agreement for this product must be signed and returned to Lifeboat Associates before shipment may be made.

Ⓘ Ⓧ This product Includes/eXcludes the language manual recommended in Sundries and Notions above.

VISA master charge

- ☐ **ZDT** — Z80 Monitor Debugger to break and examine Ⓜ registers with standard Zilog/Mostek mnemonic disassembly displays. $35 when ordered with Z80 Development Package ..... $50/$10
- ☐ **DISTEL** — Disk based disassembler to Intel 8080 or TDL/Xitan Z80 source code, listing and cross reference files. Intel or TDL/Xitan pseudo ops optional. Runs on 8080 ..... $65/$10
- ☐ **DISILOG** — As DISTEL to Zilog/Mostek mnemonic Ⓜ files. Runs on Z80 only ..... $85/$10
- ☐ **XASM-68** — Non-macro cross-assembler with nested conditionals and full range of pseudo operations. Assembles from standard Motorola MC6800 mnemonics to Intel hex ..... $200/$25
- ☐ **XASM-65** — As XASM-68 for MOS Technology MCS-6500 series mnemonics ..... $200/$25
- ☐ **TEXTWRITER III** — Text formatter to justify and pagi-Ⓜ nate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil***** prepared files ..... $125/$20
- ☐ **POSTMASTER** — A comprehensive package for mail Ⓜ list maintenance that is completely menu driven. Features include keyed record extraction and label production. A form letter program is included which provides neat letters on single sheet or continuous forms. Compatible with NAD files. Requires CBASIC-2 ..... $150/$15

New Features

- ☐ **WHATSIT?****** Interactive data-base system using associative tags to retrieve information by subject. Hashing and random access used for fast response. Requires CBASIC-2 ..... $125/$25
- ☐ **XYBASIC** Interactive Process Control BASIC — Full disk BASIC features plus unique commands to handle bytes, rotate and shift, and to test and set bits. Available in Integer, Extended and ROMable versions.
  Integer Disk or Integer ROMable ..... $295/$25
  Extended Disk or Extended ROMable ..... $395/$25
- ☐ **SMAL/80** Structured Macro Assembled Language — Package of powerful general purpose text macro processor and SMAL structured language compiler. SMAL is an assembler language with IF-THEN-ELSE, LOOP-REPEAT-WHILE, DO-END, BEGIN-END constructs ..... $75/$15
- ☐ **SELECTOR III-C2** — Data Base Processor to create †† and maintain multi Key data bases. Prints formatted Ⓜ sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source ..... $295/$20

Lower Price!

- ☐ **GLECTOR** — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 52K system ..... $250/$25

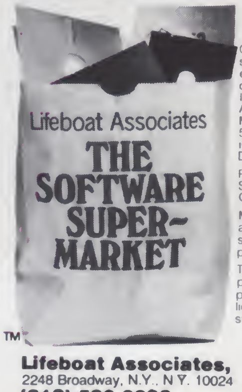
NEW!

- ☐ **CPM/374X** — Has full range of functions to create or re-name an IBM 3741 volume, display directory information and edit the data set contents. Provides full file transfer facilities between 3741 volume data sets and CP/M files ..... $195/$10
- ☐ **BASIC UTILITY DISK** — Consists of: (1) CRUNCH-14 Ⓜ — Compacting utility to reduce the size and increase the speed of programs in Microsoft BASIC and TRS-80 BASIC. (2) DPFUN — Double precision subroutines for computing nineteen transcendental functions including square root, natural log, log base 10, sin, arc sin, hyperbolic sin, hyperbolic arc sin, etc. Furnished in source on diskette and documentation ..... $50/$35
- ☐ **THE STRING BIT** — FORTRAN character string han-Ⓜ dling. Routines to find, fill, pack, move, separate, concatenate and compare character strings. This package completely eliminates the problems associated with character string handling in FORTRAN. Supplied with source ..... $45/$15
- ☐ **BSTAM** — Utility to link one computer to another also Ⓜ equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansions to send ★.COM, etc. 9600 baud with wire. 300 baud with phone connection. Both ends need one. Standard and Ⓜ versions can talk to one another. Compatible TRSDOS version also available $150/$5

★ ★ ★ ★ ★ ★

SUNDRIES & NOTIONS

- ☐ **HEAD CLEANING DISKETTE** — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5" or 8" ..... $20 ea./$45 for 3
- ☐ **FLIPPY DISK KIT** — Template and instructions to modify single sided 5¼" diskettes for use of second side in single sided drives $12.50
- ☐ **FLOPPY SAVER** Protection for center holes of 5¼" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool, tough 7-mil mylar reinforcing rings. Installation tools and rings for 25 diskettes $14.95
  Re-orders of rings only $7.95
- ☐ **PASCAL USER MANUAL AND REPORT** — By Jensen and Wirth. *The standard textbook on the language.* Recommended for use by Pascal/Z, Pascal/M and Pascal/MT users $8
- ☐ **THE C PROGRAMMING LANGUAGE** — By Kernighan and Ritchie. The standard textbook on the language. Recommended for use by BDS C, tiny C, and Whitesmiths C users $12



Orders must specify disk systems and formats e.g. North Star single, double or quad density, IBM single or 2D/256, Altair, Helios II, Micropolis Mod I or II, 5¼" soft sector (Micro iCOM/SD Systems Dynabyte), etc

Prices F.O.B. New York. Shipping, handling and C.O.D. charges extra

Manual cost applicable against price of subsequent software purchase

The sale of each proprietary software package conveys a license for use on one system only

Shopping List No.10

Everything on Shopping List #10 runs on 64K TRS-80 Model II

Software for most popular 8080/Z80 computer disk systems including **NORTH STAR, iCOM, MICROPOLIS, DYNABYTE DB8/2 & DB8/4, EXIDY SORCERER, SD SYSTEMS, ALTAIR, VECTOR MZ, MECA, 8" IBM, HEATH H17 & H89, HELIOS, IMSAI VDP42 & 44, REX, NYLAC, INTERTEC, VISTA V80 and V200, TRS-80 MODEL I and MODEL II, ALTOS, OHIO SCIENTIFIC, DIGI-LOG** and **IMS 5000** formats.

™The Software Supermarket is a trademark of Lifeboat Associates

Lifeboat Associates,
2248 Broadway, N.Y., N.Y. 10024
(212) 580-0082 Telex: 220501

Chesney E. Twombly
15 Storer Street
Kennebunk ME 04043

8080 Program Loader/Relocator

The "CONOPS" series continues to operate with this latest entry.

```
                         TTL    TSC 8080 RELOCATOR MODIFICATION
                   *
                   * PROGRAM LOADER REPLACES TAPE LOADER.
                   * RELOCATES PROGRAMS DURING LOADING FROM KEYBOARD.
                   *
                   * CHANGE THE CALL INSTRUCTION AT 40F4 TO READ;
                   *    'CD E5 40         CALL ENTER1'
                   *
                         OPT    MEM,NUM,PAG
428A                     ORG    $428A
                   *
              6C1D INCHR    EQU    $6C1D
              4000 STACK    EQU    $4000
              6C78 ADDOUT   EQU    $6C78
              6C28 CONVT    EQU    $6C28
              6C40 BYTE     EQU    $6C40
              6C86 OUTS     EQU    $6C86
              4450 OLDPTR   EQU    $4450
              445A OFFSET   EQU    $445A
              4452 OBJEND   EQU    $4452
              4272 CMPDHB   EQU    $4272
                   *
                   * PROGRAM LOADER
                   *
428A 2A 50 44      LOAD     LHLD   OLDPTR
428D EB                     XCHG
428E CD 78 6C               CALL   ADDOUT
4291 2A 5A 44               LHLD   OFFSET
4294 19                     DAD    D
4295 E5                     PUSH   H
4296 C1                     POP    B
4297 CD 1D 6C      LOAD2    CALL   INCHR
429A FE 7F                  CPI    $7F;'DEL'
429C C2 A7 42               JNZ    CHKADR
429F 0B                     DCX    B
42A0 1B                     DCX    D
42A1 CD 78 6C               CALL   ADDOUT
42A4 C3 97 42               JMP    LOAD2
42A7 FE 4C         CHKADR   CPI    'L'
42A9 C2 B2 42               JNZ    LOAD3
42AC CD 78 6C               CALL   ADDOUT
42AF C3 97 42               JMP    LOAD2
42B2 CD 28 6C      LOAD3    CALL   CONVT
42B5 CD 40 6C               CALL   BYTE
42B8 02                     STAX   B
42B9 CD 86 6C               CALL   OUTS
42BC 2A 52 44               LHLD   OBJEND
42BF CD 72 42               CALL   CMPDHB
42C2 D8                     RC
42C3 03                     INX    B
42C4 13                     INX    D
42C5 7B                     MOV    A,E
42C6 E6 07                  ANI    7
42C8 FE 00                  CPI    0
42CA CC 78 6C               CZ     ADDOUT
42CD C3 97 42               JMP    LOAD2
                   *
42D0                        DS     $38
                   *
4308 E5            PCRLF    PUSH   H
                   *
                            END
```

Listing 1. TSC 8080 Relocator modification.

Microcomputers are likely to have blocks of memory reserved for ROM operating systems, thereby excluding those addresses from other uses. The Heath H8 computer provides a good example: the ROM panel monitor, PAM-8, occupies hex addresses 0000–0400. Block 0400–2000 is reserved for uses that are undefined by the manual. RAM addresses 2000–2040, as well as 80 bytes at some higher location, are required for PAM-8 use. The H8 console driver, a video terminal interface, uses RAM block 2040–2163. The standard H8 system has no RAM below hex address 2000.

There seems to be an abundance of 8080 machine-language programs available in books and magazines or from software vendors and users' groups. Published listings of such programs are likely to be unusable until relocated. Hand relocation, performed mentally during program entry, is the obvious, albeit tedious and error-prone, method.

The TSC 8080 Relocator, a copyrighted product of Technical Systems Consultants, PO Box 2574, West Lafayette IN 47900, can read a program on Intel ASCII hex format paper tape and relocate it to any specified address. It will relocate a program already stored in RAM but does not allow the option of loading from a keyboard. This relocator is a marvelous program and worth many times its $8 price.

Modification

Listing 1 replaces the paper tape reader of the TSC relocator with a keyboard loader that can relocate a program while it is being loaded. This modification, though designed for the Heath H8, can be used on any 8080 system. It uses several subroutines of the H8 Console Operating System, CONOPS, published in the July 1979 issue of *Kilobaud Microcomputing*, p. 108.

First load the unaltered TSC program into RAM starting at hex address 4000. When it is working without errors, you can load the modification at the addresses shown in the listing. Non-Heath users must enter the ADDOUT, CONVT, BYTE and OUTS subroutines from the listing in the CONOPS article at an appropriate location. The address parts of all instructions calling these subroutines must be changed.

One change in the original program must be made before this modification will function properly. Change the CALL instruction at 40F4 to read CD E5 40 CALL ENTER1.

ADDOUT	6C78	BYTE	6C40	CHKADR	42A7	CMPDHB	4272	CONVT	6C23
INCHR	6C1D	LOAD	428A	LOAD2	4297	LOAD3	42B2	OBJEND	4452
OFFSET	445A	OLDPTR	4450	OUTS	6C86	PCRLF	4308	STACK	4000

Symbol table.

When using the modified relocator, you can activate the loader and display the hex address of the first bytes of the program being loaded by typing Y in response to the prompt LOAD FROM TAPE. As each byte is entered, an offset is added to place the code in RAM at the desired destination address, and your video monitor will display the byte as two ASCII characters.

Typing L will display the next address for verification. RUBOUT will display the last address entered and, in effect, is a back-space function. Both commands can be entered only after entry of a complete hex byte.

Upon entry of the final byte, the END ADDRESS you entered at the start of the program, the statement, LOAD COMPLETED, will be displayed, and the relocator will pause until you type a space. Follow the instructions given in the Relocator manual to complete the load/relocate operation. ■

✓Reader Service—see page 257

Ralph Tenny
PO Box 545
Richardson TX 75080

Get Started with MicroStart

MicroStart makes a dandy tool for experimenting with CPU chips.

Now that you have your computer system running, do you find yourself waiting for the games, etc., to be finished? Do you have to send the kids to bed early so you can "play" with your own computer? Perhaps the machine is automating your house—fire sensors, security alarms, etc.—so you are reluctant to take it offline just to tinker with some experimental hardware.

Don't rush out to buy another whole machine—build MicroStart! This little board is one solution to the problems mentioned above, but it also enables me to experiment with other microprocessor chips besides the MCS6502, which comes in my KIM-1 system.

MicroStart is really only one-third of a computer—the memory. Fig. 1 shows a block diagram of a computer that consists of a CPU (central processing unit), memory and I/O (input/output) circuitry. MicroStart is different from other computer memory boards because M/S is a stand-alone memory which can be loaded with data that can be used for any imaginable purpose.

How the Memory Works

MicroStart's stand-alone feature is the result of adding five push-button switches (see Photo 1) and some support circuitry to accomplish data entry. Another MicroStart difference is that full control of both memory and data is achieved with only those five switches: CLR, INC, M, L and RL.

Here's how they work: After power is applied, pressing CLR forces both the data bus and the address bus to zero (00000000_2—note that both address and data are 8-bit values). Switches M and L control 4-bit counters that are driven by a slow clock. This allows the operator to hold down either button until the counters contain the desired data value.

This data is displayed by the row of LEDs that appear along the top of the M/S board (see Photo 1). These LEDs are in four groups of four each, representing (left to right) most significant data nibble, least significant data nibble (a nibble is four bits—half a byte) and eight bits of address data. After CLR is pressed, all LEDs will be off.

Let's assume that the data required in memory address 00_{16} (all eight right-hand LEDs off) is $A5_{16}$. Hold down RL and M until 1010_2 appears in the left-hand group of LEDs. Then hold down RL and L until the second group of LEDs shows 0101_2. The required data is now entered in the memory at location 00_{16} and can be verified any time later by pressing CLR so that the address counter returns to 00_{16}.

The RL (RAM load) key worked this way during the operation described above: As

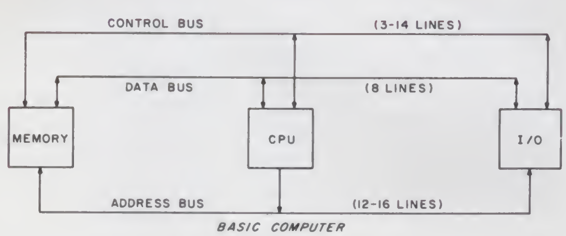


Fig. 1. Basic computer.

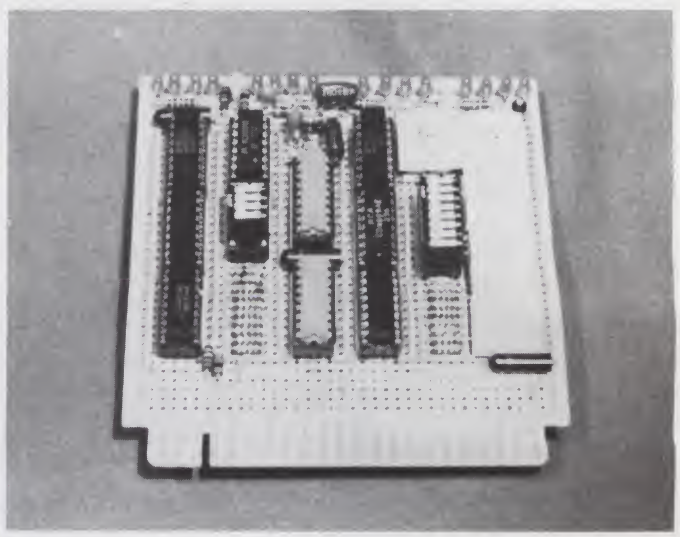


Photo 1. MicroStart is a 256 byte memory with five control keys that load data into memory.

long as the RL key is held down, the RAM (Random Access Memory—IC1 and IC2) is loaded about 10 times per second with the data on the Data Generator output lines. Thus, the data showing on the LEDs is *always* the memory contents at the address shown by the address LEDs. A faulty memory location will remain unchanged, so the load operation also checks the memory for proper operation.

After data is entered into location 00_{16}, press INC once and the LED on the far right will show address 01_{16}. Again, the proper data is entered using RL, M and L keys, and then INC is pushed to access location 02_{16}. In this fashion, data is entered into as much of the memory as is desired.

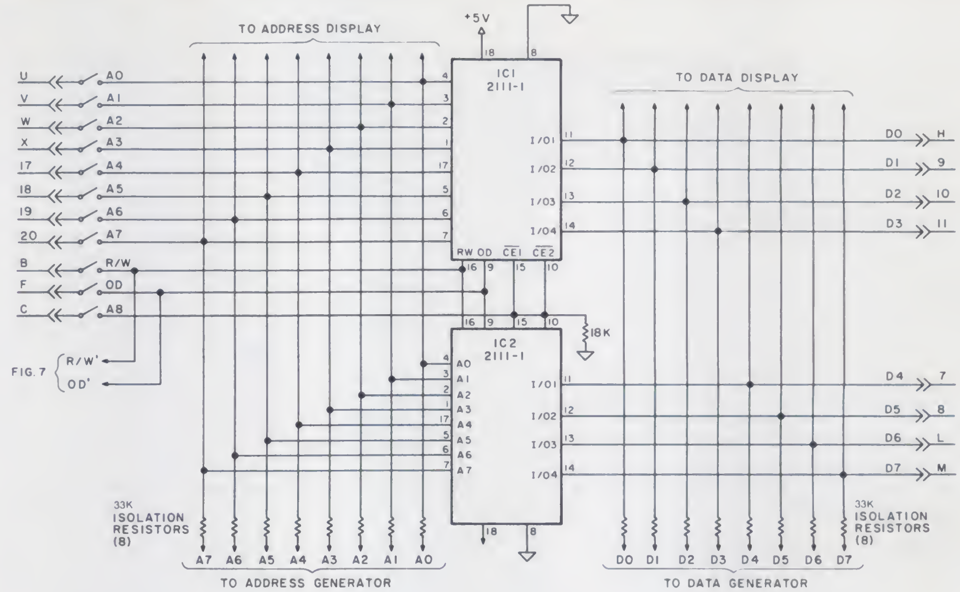


Fig. 2.

After all the data is entered, what good is it? Well, since MicroStart is a stand-alone memory, the data *could* represent eight switches controlling some gadget. Whenever a new condition is needed, simply increment the memory to the next programmed location. However, suppose there was another board available with a microprocessor and some I/O circuitry. By hooking the two boards together, the diagram of Fig. 1 is completed and you have a small computer! Which microprocessor? That's up to you. Photo 2 shows a National SC/MP CPU board, and Photo 3 shows a Signetics 2650 CPU board.

Now that we have a computer, what can we do with it? How about a darkroom timer that can be programmed to "remember" and measure several times—one for each of several photo-developing operations—and display time remaining (countdown to finish) for each operation. How about interactive games? A very interesting one, PNG-PNG, is discussed later. Would you like to measure the reaction times of your friends? That one is easy—light an LED as a signal and measure how long it takes for the operator to press a button in response.

The possibilities are endless, but note that the original M/S was built to test various microprocessors under similar circumstances as is discussed below. Once MicroStart was working, experiments with different microprocessors cost me less than $30 for each new machine.

Naturally, if you have been using canned programs or writing your own in BASIC or some other high-level language, an education is awaiting you. You really have to get down to the basics—assembly language and machine language—with MicroStart. For me, the software challenge is almost as exciting as creating new hardware; the ultimate for me is to create a system (software *and* hardware) better than the last one. I guess that is why MicroStart has been so much fun.

The Circuitry

Fig. 2 shows the memory portion of M/S. Note that each line out of the memory chips has two branches. The main branch passes through isolation re-

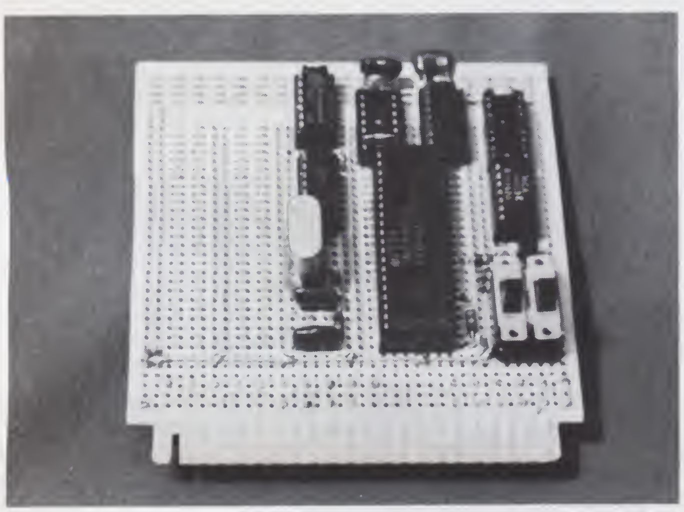


Photo 2. A CPU board built around the SC/MP microprocessor. Two switches control the reset conditions. Note keyway on card edge and compare key location with Photo 1.

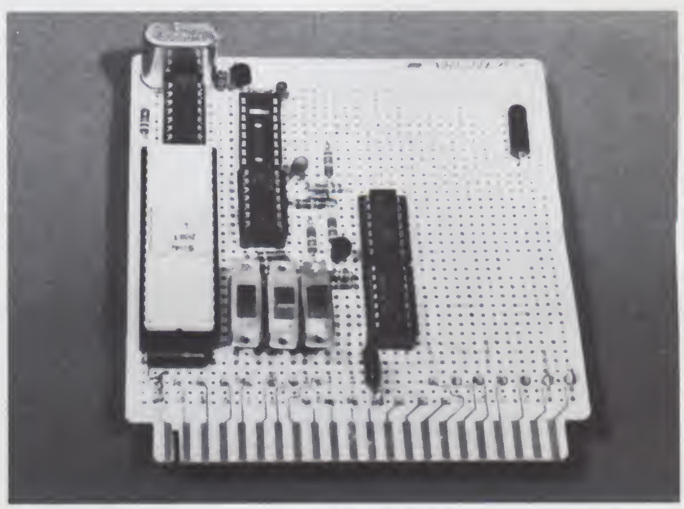


Photo 3. CPU board built around the Signetics 2650 microprocessor. Note that three switches are required to control reset conditions.

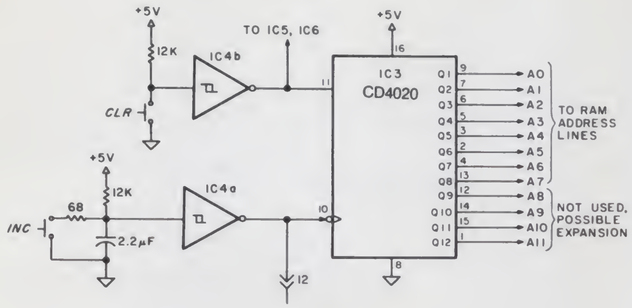


Fig. 3.

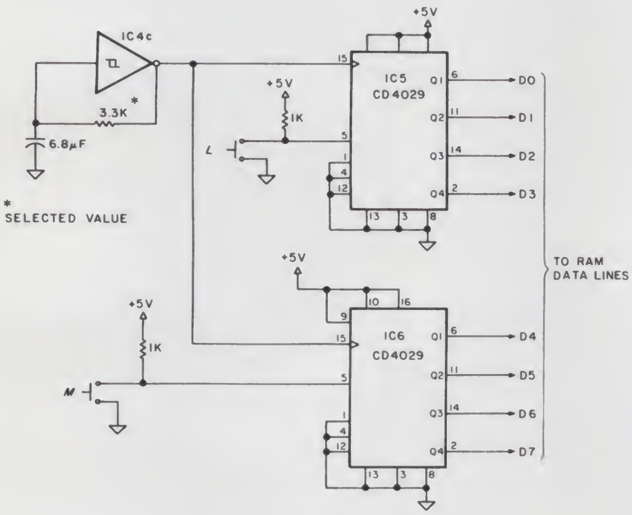


Fig. 4.

sistors to the edge connector, where it is connected to the CPU board. All of these lines except A8 (the 9th address line) pass through the resistors.

Fig. 3 shows the local address generator. IC3 is a ripple-carry counter that generates 12 bits of binary address—enough for 4096 bytes of memory. Although the prototype MicroStart uses only eight lines (256 bytes), it can easily be expanded to use two 1024 × 4 RAMs for a full kilobyte of memory. No other changes will be needed and most 1K × 4 memory chips will fit the same sockets used on MicroStart.

In the prototype M/S, address line A8 is grounded through a resistor and holds the chip-select lines of IC1 and IC2 low (enabled) except when the CPU board might pull it high. This connection will allow more memory to be added to the CPU board if that should be needed. That is, when A8 goes high, IC1 and IC2 are disabled, and other memory can then be addressed.

The addresses are generated by IC3 under control of two switches—INC and CLR—which work with sections of IC4. IC4A is a Schmitt trigger with two resistors and one capacitor to debounce INC. IC3 is advanced one count at a time by the output from IC4A. The CLR switch drives IC4B to reset both the address and data counters to zero; no debouncing is necessary since a bounce merely repeats the reset action.

Fig. 4 shows the local data generator, consisting of two counters and a free-running oscillator. The M switch is held closed to set up the four most significant data bits, and L sets the lower four data bits. IC5 and IC6 are pre-settable up-down binary counters, which are advanced individually when switches M and L are closed. The oscillator is IC4C, which drives the clock inputs of each counter. Pin 5 of the counter is called Clock Inhibit, which prevents counting whenever the line is high. When either M or L is closed, the associated counter advances one count each time IC4C clocks.

Naturally it is necessary for the actual data and address values to be displayed for operator feedback; Fig. 5 shows the display drivers. Three CMOS hex buffers "read" the address and data bits as they appear on the memory lines, and each section drives one LED.

Note that no series resistors were used between the CMOS buffers and the LEDs on the prototype board. This is permissible only under certain conditions and is possible because the CMOS output is inherently current-limited. Operating into a short circuit such as an LED or transistor base is permissible only if the V_{cc} is 5 volts or less, and if the external device is able to withstand the short-circuit current output of the CMOS device. It should also be noted that any CMOS device operated into a short circuit will not develop the normal logic levels produced at normal load levels.

Fig. 6 shows an alternate display design—four decoders driving seven-segment displays to produce hexadecimal readout of data and address. This display design was considered for the prototype, but was rejected in keeping with the desire to keep MicroStart on a small board. The seven-segment readout would be entirely appropriate if MicroStart were installed in a permanent chassis.

Figs. 7, 8 and 9 explain a very important section of MicroStart—the RAM-load circuitry. First it is necessary to under-

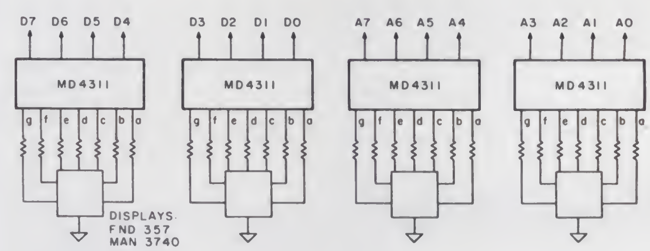


Fig. 6.

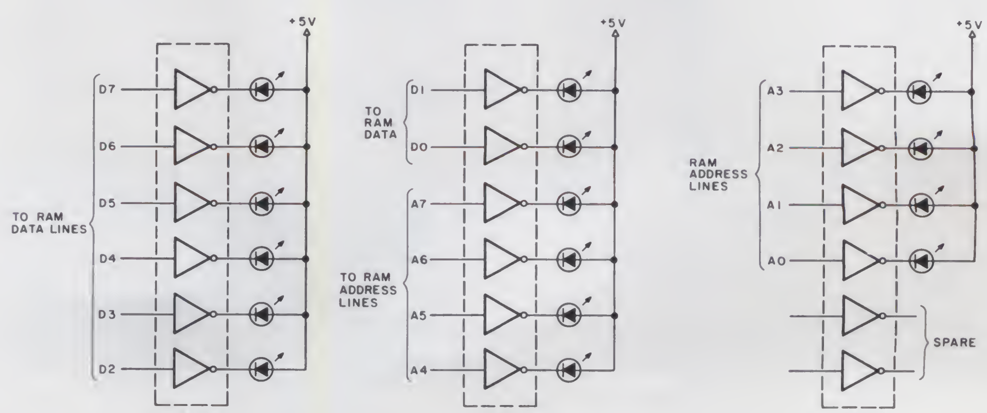


Fig. 5.

stand how the 2111-1 memory chip functions (Fig. 7). Since the four data inputs and the four data outputs share the same package pins, the input and output operations must be separated. When the OD line is high, the output lines are disabled. If either $\overline{CS}$ line is high, both input and output lines will be disabled and the four data output pins will be floating. This permits any reasonable number of 2111s to share the same data bus without causing any appreciable loading effect.

If both $\overline{CS}$ lines are pulled low along with OD, while R/W is high, the data pins will output the contents of the memory cell addressed by the eight address lines. With OD high and both $\overline{CS}$ lines low, data will be written into the addressed memory cells whenever R/W goes low for some minimum time (380 ns for the 2111-1) and then returns high. To summarize, both $\overline{CS}$ lines must be low for any memory operations to take place. OD must be low for data output; high for data entry. R/W must go low and return high for data entry.

MicroStart uses a sort of semiautomatic data entry sequence. The circuitry is shown in Fig. 8 and the timing diagram in Fig. 9. A separate oscillator (IC4D) generates a basic timing waveform that governs the data entry sequence. IC4E detects the falling edges of IC4D's output and generates a pulse approximately 40 us long. This pulse raises OD to disable the data output. IC4F detects the rising edge of IC4E's output and generates an R/W pulse approximately 25 us long. Thus, about 10 times each second, data from the data generator is written into memory.

In Fig. 8, R1 holds OD low except when it is pulled high by IC4E or by the CPU (via card-edge connector shown in Fig. 2). Diode D1 disconnects OD from IC4E whenever the CPU drives the line. Switch SW1 is opened during CPU operation to prevent accidental data entry via the control keys. This feature has proven its worth several times when my young grandson has "helped" me! RL is the RAM load switch mentioned above. Like D1/R1, the D2/R2 pair serves to keep R/W high except during data entry and to disconnect IC4F during CPU operation.

As mentioned before, OD is low about 99.5 percent of the time, and only RAM output data ever shows on the display LEDs. With OD low, the RAM is in its low impedance state, so that data from IC5 and IC6 cannot affect the display or the RAM I/O pins. When OD is high, the RAM input lines are high impedance but will read data to be entered by R/W.

Construction

Build and test MicroStart in sections. For example, build all sections of IC4's circuitry and apply power. The resistor and capacitor values shown for IC4C are approximate and should be varied for individual reaction times as discussed below. Use a logic probe or oscilloscope to verify that each section is working; IC4C and IC4D will be free-running, while the other sections function only when an associated switch is operated.

Next, add IC5 and IC6 with their associated circuitry, including buffers and LEDs. Adjust the RC values of IC4C for a comfortable rate in this manner: Pick a data value ($A5_{16}$, for example) and operate M and L until the display reads 1010 0101. If IC4C is oscillating too fast, it will be difficult to release M or L when the data is exactly correct. A slow rate will make it seem like *forever* before the bit pattern is correct. A nice compromise is a rate slightly too fast for comfort; allow the data to count up almost to the desired value, then release the key and press it just long enough to catch single pulses.

The display will "run up" until almost correct, then two or three single pulses will set it correctly. The learning curve is short, and this five-button layout is much simpler than a 16-key (hexadecimal) keyboard with INCrement, Enter and Clear switches added. Continue the checkout by being sure that IC3 can be controlled by INC and CLR, then wire sockets for the memory and demonstrate that the memory can be loaded and read back under your control.

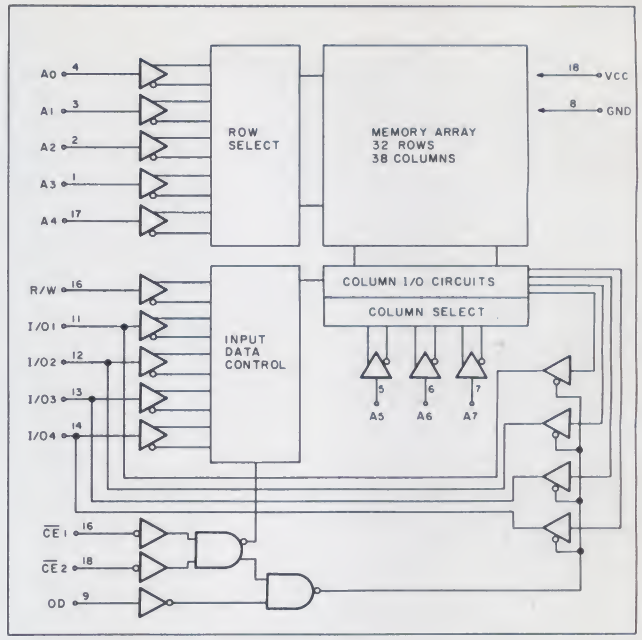


Fig. 7. Block diagram of the 2111 memory, organized 256 × 4. Two 2111s are required for MicroStart.

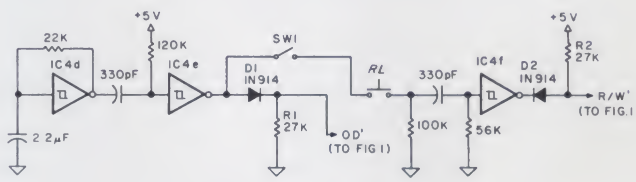


Fig. 8.

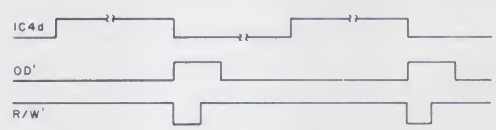


Fig. 9.

If a faster method of data entry than the M/S five-key setup is desired, IC5 and IC6 can be replaced with quad latches driven by a hexadecimal keyboard. This is particularly appropriate if MicroStart is to be installed in a permanent chassis. At that point, it would also be advantageous to have the seven-segment display setup mentioned before (Fig. 6). M/S would then be an even more convenient tool for developing new hardware and software without disrupting some other computer setup.

The CPU Board

The following information will be mostly guidelines—MicroStart is too flexible in application to be tied down to one man's ideas! First, decide on a project goal. The CPU board configuration will depend upon which microprocessor is to be tested. All of them have different features and capabilities, but certain features must be known and understood. The important features are:

1. Which signals disable the processor so that the data bus is quiescent.
2. Which signals allow the processor to be temporarily

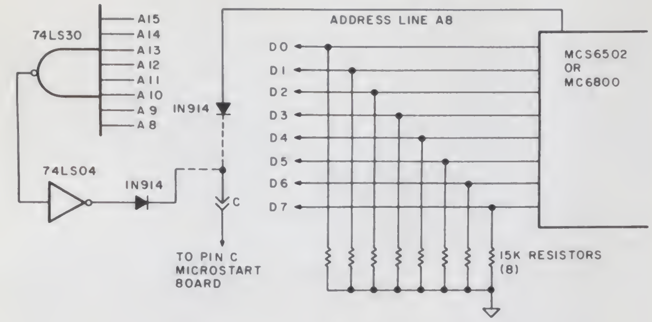


Fig. 10.

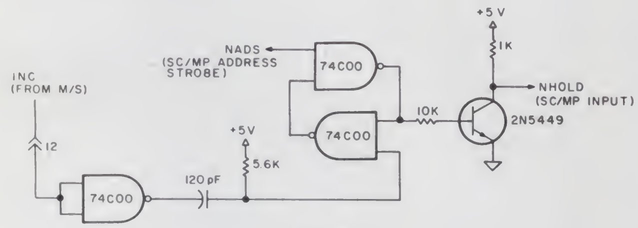


Fig. 11.

stopped so that single-step operation is possible.

3. Which address the processor accesses first after reset, and what type of data the processor expects to find there.

4. Whether the processor can set its address lines to high impedance.

The four items above affect the design of the CPU board directly. If item 4 turns out that the processor can set its address lines to high impedance during reset, the isolation switches in the address lines of M/S are not needed. If the processor does not access address 00_{16} after reset (item 3), appropriate circuitry must be built to force this access.

For example, the MC6800 accesses $FFFE_{16}$ and $FFFF_{16}$ after reset, and the MCS6502 reads $FFFC_{16}$ and $FFFD_{16}$ after reset. Both machines treat the data there as an indirect address; that is, they use the data as an address to find the first instructions to execute. If the data bus can be forced to 00 each time the 6800 or 6502 executes the first two fetches after reset, then each machine will jump to address 0000_{16} for its first instruction. From that point on, both machines will act like the processors that reset to 00_{16}.

Fig. 10 shows one method of forcing the data bus to read 00. Address line A8, which disables the memory on M/S whenever it is high, is normally driven directly by the processor. In this case, A8 is diode-coupled to both the processor and an address decoder. The 74L830 senses the eight most significant address bits, and via an inverter, address line A8 is driven high each time the microprocessor tries to access any address above $FFEF_{16}$. Since A8 disables the memory so its output lines are high impedance, the resistors on the data lines pull them low. It should also be noted that the CLR button must be used to set the M/S address generator to 00 so that it will not interfere. Thus, each time the 6800 or 6502 reads the reset address, it receives data directing it to where the memory on M/S starts.

In dealing with items 1 and 2 above, each CPU board must have a switch to force the reset condition on the microprocessor. Normally, reset is a momentary switch, but since data entry into MicroStart is a lengthy process, a toggle switch is better.

Fig. 11 shows one solution to the single-step operation of the SC/MP; this was patterned after data furnished in the SC/MP operator's manual. NADS is an SC/MP strobe that occurs just after the start of a machine cycle, and it sets the latch in Fig. 11 so that the NHOLD line is pulled low to stop the machine. The latch is reset by the operation of the INC switch on M/S, which allows the processor to execute the next part of the machine cycle. Similar arrangements exist for most other microprocessors, but each is likely to be somewhat different.

To the operator, the effect of the NHOLD line is that each memory fetch can be observed on the address and data LEDs of MicroStart. Not only is this instructive in itself, but also it becomes possible to scale the microprocessor's speed to "people" speed. If the program has errors, it is possible to see the program execution go astray. It is then possible to revert to the data load operation and make program corrections.

A number of microprocessors (SC/MP, 2650, 1802) have Flag outputs—single-bit output ports that can be set under program control. These can be utilized in several ways, both in regular operation and in program debugging. For instance, SC/MP has three Flags: F0, F1 and F2. If F0 is connected to a transistor that pulls down on NHOLD as shown in Fig. 12b, the machine will stop whenever the flag bit is set.

Fig. 12a shows a two-instruction sequence that activates Flag F0 and stops the machine. This is very useful when long programs are entered, since

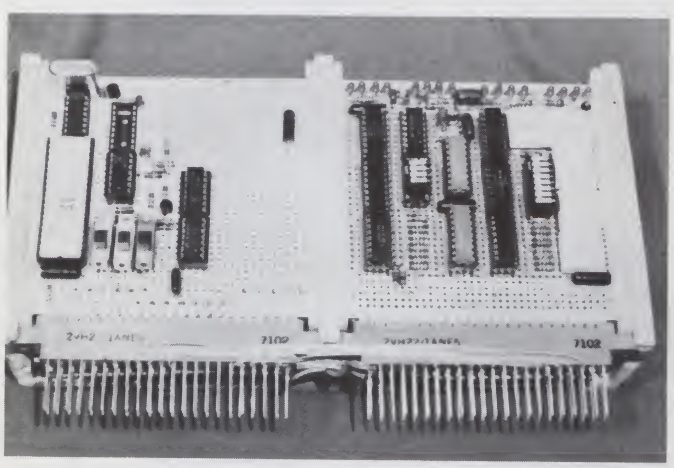


Photo 4. MicroStart and a CPU board plugged into a common chassis, ready for business!

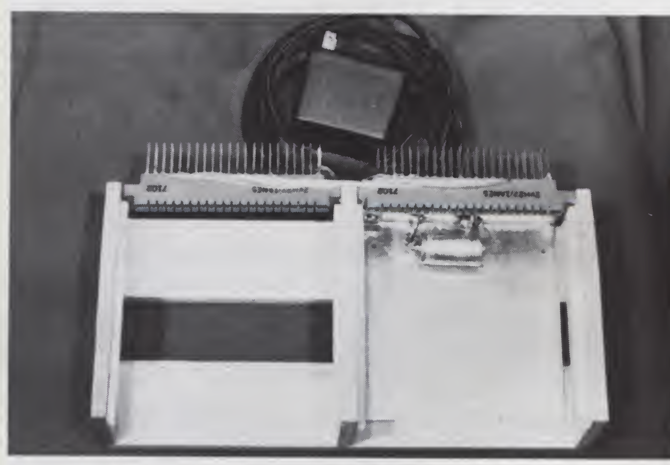


Photo 5. View of empty chassis with ac adapter power supply. Chassis built from wood strips and assembled with Super Glue.

Photo 6. View of chassis construction. End piece is notched for connector lug, and holes are drilled to accommodate press-in lugs of the card guide.

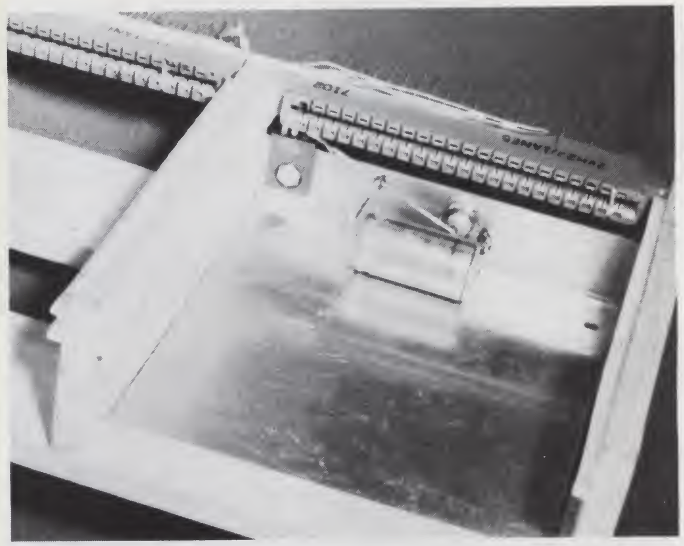


Photo 7. Close-up of filter capacitor and three-terminal regulator. RCA-type jack used to bring in power from ac adapter. Note keying strips in edge of connectors to prevent plugging boards into wrong slots.

LOCATION	CODE	ASSEMBLY LANGUAGE	COMMENTS
16	C4 01	LDI 01	LOAD ACCUMULATOR WITH 01_{16}
18	07	CAS	COPY ACCUMULATOR TO STATUS REGISTER (SETS FLAG F0 HIGH)

(a)

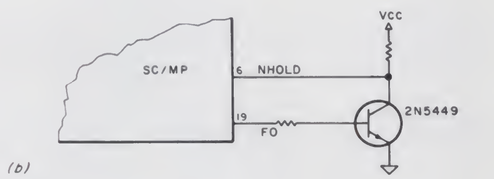


Fig. 12.

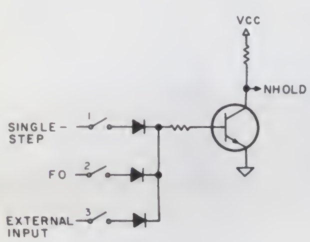


Fig. 13.

small pieces of the program can be loaded *and tested* before more program is added. That way, any error is located as soon as it is entered and can be corrected.

After each program segment checks out, simply write over the three "stop" instructions with new code. The single-step mode and stop mode can be combined as shown in Fig. 13 along with other signals to stop the processor. If switch 1 is closed, single-step operation results, while switch 2 allows a programmed stop and switch 3 allows an external event to stop processing.

One other circuit addition has proven universally valuable—phantom strobes. If some unused address line is ANDed with lower order address lines as shown in Fig. 14, discrete signals can be produced with a single line of code. For example, LDA ENA1 (an assembly-language statement) will cause the processor to access the address of the phantom location named ENA1. This address has been defined by hardware connections as 0901_{16}.

Fig. 15 shows this address in binary form and identifies the particular bits. It can be seen that address lines #11, 8 and 0 are high. A11 and A0 are decoded by gate A in Fig. 14 and cause the line ENA1 to go high, while A8 disables the memory on MicroStart to avoid disturbing normal memory. Such strobes can be used to turn some external circuit on or off, or they can enable another section of memory or a latch to store special data.

Note this special caution: Because address bits A10, A9 and A7 through A1 are not decoded, any odd address between 0901_{16} and 1000_{16} will also enable ENA1. Similarly, many addresses within this same range will enable ENA2, ENA4 and ENA8. This is acceptable so long as the memory space (the total amount of memory any particular CPU can address) in that address range is not needed. When address space is almost used up, decoding for any particular address must become non-ambiguous.

Another handy feature that has been incorporated in the CPU boards used with MicroStart has been an I/O plug. This is simply a 16-pin IC socket with a variety of different signals wired to it as detailed in Fig. 16. Two sockets are shown—one for the SC/MP and one for the 2650. Both sockets have the 8-bit data bus brought out from the processor, and each has some of the phantom strobes.

The 2650 has only one Sense line (a one-bit input port) and one Flag. Both Sense and Flag are brought out from the 2650, along with four strobes, the processor Interrupt line and a debounced input line to the Sense port, which allows a switch input to the Sense line.

The SC/MP has an abundance of interesting lines—the eight data lines, two strobes, the processor Read and Write strobes (NRDS and NWDS), SIN and SOUT (serial input and output ports), one of two Sense lines and one of three Flag lines—so the choices were harder. These output *plugs* allow the use of ribbon cable to

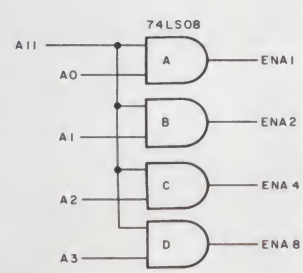


Fig. 14.

BIT	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
BINARY	1	0	0	1	0	0	0	0	0	0	0	1
HEXADECIMAL		9				0				1		

Fig. 15.

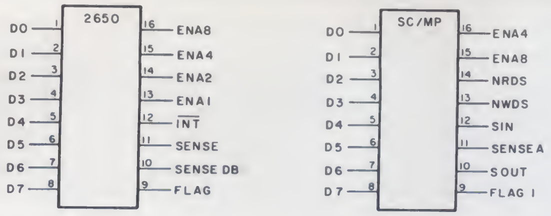


Fig. 16.

connect an experimenter chassis (Photo 8), which greatly eases building and testing experimental hardware.

One particular caution must be heeded: All outputs that come directly from the microprocessor are driven by MOS transistors and are therefore sensitive to voltage spikes and static electricity. If you zap a Flag or Sense line, the processor may still be usable, but even one zapped data or address line puts you out of business! Use special care when hooking up, wire the test circuits while the ribbon cable is disconnected *at the CPU board* and be sure to have the power and ground connections made between the CPU board and the test board before hooking up the ribbon cable. Apply power to MicroStart and the CPU board, initiate reset and do all programming after the test hookup is complete.

Turn on the Computer!

Let's assume that a program has been loaded as was described earlier and that a CPU board is plugged in, waiting to be taken out of reset. The CPU should then execute the program you entered, assuming that the CPU will reset to 00_{16} (or has been forced to 00_{16} as outlined above) and assuming that the program didn't have too many errors. Errors in a computer program cause one of two effects—the program runs but does other than was intended, or the program "bombs"—that is, does not run.

The distinction is important; if the program runs until it is stopped, finding the error is much easier than if the program bombs. In a program that bombs, the errors cause the CPU to vector off into unprogrammed memory or into memory space where no memory is installed. Otherwise, the program may wipe itself out, step by step, until the CPU shuts itself off. Either way, the CPU must be reset to gain control. The program that runs, but not properly, simply performs different actions than intended or fails to perform the intended task. Obviously, the *correct* program performs exactly as expected.

If a program bombs, or if it simply isn't quite right, double-check the program logic and the correctness of the code you intended to enter. Be sure that you understand what each machine instruction is supposed to do. Finally, reset the CPU and check the program entry to be sure that each data byte is correctly entered.

On programs that bomb, pay particular attention to branch calculations; jumping one byte too far or one byte short causes the CPU to try to execute data instead of an instruction. If this happens, the result is usually wildly unpredictable. Be sure the loop count is correct on iterative program segments. Check everything!

The Test Chassis

Almost all the connections in MicroStart and the CPU boards were wire-wrapped, except for some connections to discrete components. Both Radio Shack #276-154 and OK Machine and Tool #H-PLB-1 4 × 4½ inch prototyping boards have been used with equal success. Use large-size bus wire to connect ground and power to the on-card bus strips (Photo 9) and use a minimum of two socket pins to connect power and ground. Use several tantalum decoupling capacitors between power lines and ground lines on each board. Use large, high-quality filter capacitors on the power supply.

An excellent power source for small projects such as this is one of the several transformer-on-a-cord sets sold for line operation of battery-powered appliances. Some versions come with switch-selectable 4.5 V, 6 V, 7.5 V and 9 V outputs, while others are 9 or 12 volt units. The minimum acceptable current rating is 300 mA, but many versions have this rating. Heavier-duty units are doubtless available, but will be difficult to locate.

For MicroStart, use a minimum of 9 volts dc and add at least 1000 uF of filter capacitor. Use a three-terminal regulator on each board or use a common regulator on a heat sink as shown in Photo 6. The prototype MicroStart has an unusual chassis—a wooden frame assembled from 47¢ worth of scrap lumber and four card guides (25¢ a piece), all glued together with Super Glue! This handsome (?) device is detailed in the photos—even the heat sink is glued in!

Test Programs

What can you do with such an abbreviated computer? Don't sell small memory, simple systems short—*it takes an* amazing amount of well-written assembly-language program to fill 256 bytes of memory! Remember that M/S can be easily expanded to 1K bytes just by changing the memory ICs and hooking up extra ad-

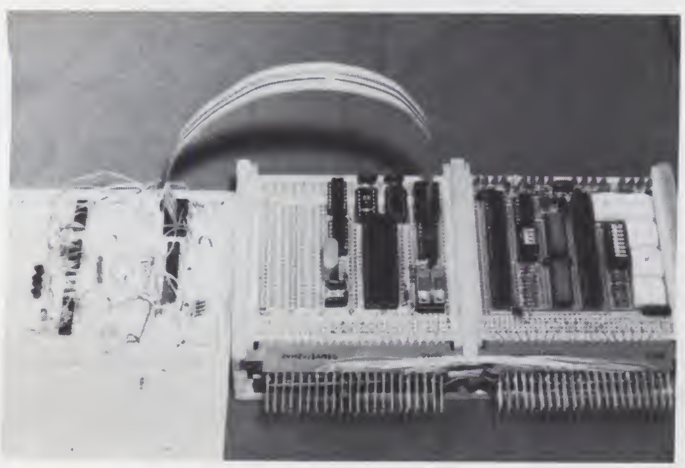


Photo 8. MicroStart at work. Ribbon connector feeds signals to experimenter strip to drive experimental hardware.

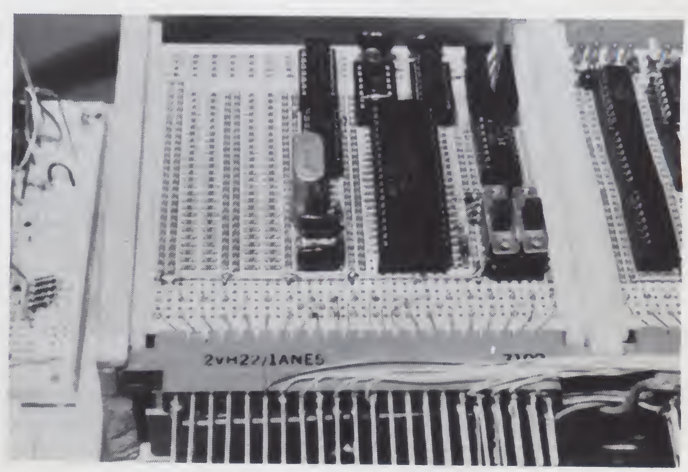


Photo 9. Close-up of SC/MP CPU board. Note large power bus and oscillator crystal.

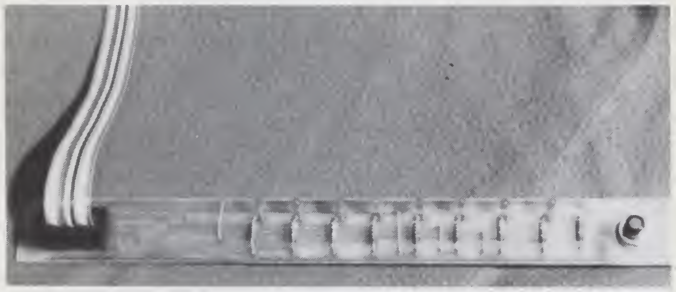


Photo 10. Ribbon cable feeds LED panel to be used for PNG-PNG (see text). "Quickie" PC board holds minimum of circuitry to drive LEDs.

dress lines and isolation switches.

Learn a new microprocessor instruction set by writing and testing simple routines such as: Move data from one area of memory to another, set up a timing loop that turns an LED on and off at 1/10 second intervals, make a counter that counts switch closures and lights a different LED each five pulses (use software to debounce the switch) and make a circuit and software to tell which of two switches close first. Then, combine these various small routines into some kind of interactive game such as one that is sometimes called PNG-PNG. A professional programmer I know often uses this game to learn about a new computer that has front-panel switches and data lights.

The general idea is to light the front-panel LEDs in sequence so the light appears to be moving back and forth across the front panel. It becomes an interactive game when the operator is required to close a switch precisely during the time either end LED is lighted. The switch closure reverses the apparent motion of the light, and closing the switch too early or too late counts against the operator. Numerous variations are possible, but one of the most entertaining is to have the computer speed up the light's movement if the operator's score is higher than the computer's.

Flowchart 1 shows one implementation of PNG-PNG. The following operational sequence is shown, assuming the use of the LED board shown in Photo 10. The action begins as the right-hand LED is turned on. After a delay (which should be about .2 seconds), a test is made to see if the switch is closed. A test is then made to see if an end LED is energized. Since this is the first pass, that answer would be "yes."

If the player had pressed a key as soon as the "game" started, the player's score would be incremented. Otherwise, the computer would gain a point. Since this was an end LED, the first move would be to move the light to the left, delay and test "Key?" and "End LED?" again. On the second test, it would not be an end LED, so the second move would again be to the left (entering a loop at Entry 2). This looping will continue until the next time the "Key?" test is true, when the program will take the Entry 3 path and make loops until the other end is reached.

In the above discussion, the "Score = Win?" test was ignored. This test is made (in this implementation) only after an "End LED?" test is true. Even though the computer gets a point whenever the key is closed (but not at an end LED), the program logic works fine. A more thorough treatment would be to test each score every time it is incremented.

One other "missing" test should be mentioned. Although a test is made for key closing, no subsequent test is made to see that the key is opened before the next test for closed switch. This will become a problem if the computer is given the ability to speed up the action—shorten the delay—unless a limit is also placed on how short a delay is permissible. Ordinarily, any "key closed?" test must be followed by a "key open" test, but the delay effectively serves to buffer the key action in this case.

PNG-PNG Input/Output for 2650

Figs. 17 and 18 illustrate the use of I/O plugs (Fig. 15), with Fig. 17 detailing output drive for the "moving LED" display and input sensing for the key and a "start" switch to be discussed later. Fig. 18 shows one way to create an optional display to show the scores at game end.

The output portion of Fig. 17 simply uses the 8212 as an 8-bit

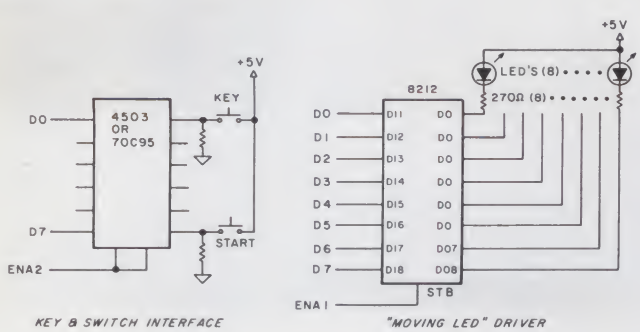
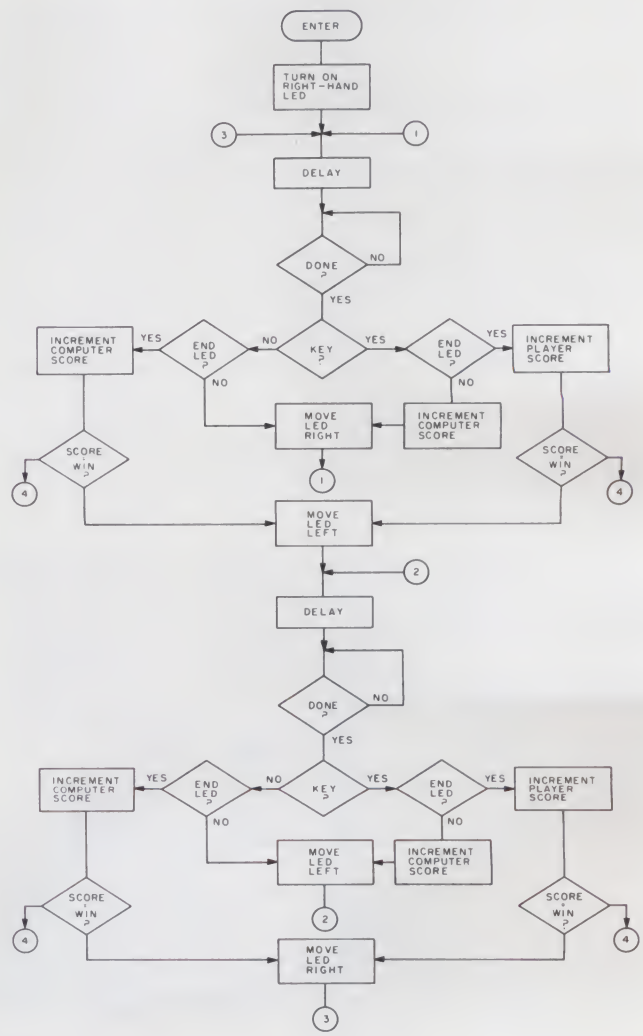


Fig. 17.



Flowchart 1.

✓Reader Service—see page 257

latch that drives the selected LED until another LED is selected by writing a new data word to the latch. On the input side, each switch connects to one section of a Tri-state buffer, with the two output lines feeding Bit 0 and Bit 7. Note that address selection is accomplished by using ENA1 and ENA2 (Fig. 14).

Fig. 18 shows how to display one-digit scores for the computer and player. The 4511 is a CMOS latch/LED driver that decodes BCD input data into seven-segment drive for common-cathode LED displays. The player score is fetched and then written to ENA4, while the computer score is written to ENA8. Note that this display restricts the score to a maximum of nine; by using two more 4511s and two more LED displays, score display can be expanded to 99 maximum. Use the same address strobes, but load the second 4511 in each display from data bits D4 through D7.

The flowcharts we will discuss next are simply program segments that perform the specified function. Since the 2650's instruction set is unusual, suggested assembly-language statements accompany various blocks of the flowcharts. The 2650 has two sets of three registers, plus a Register 0. The flowcharts are written with the assumption that Register 1 will contain the computer score, Register 2 holds the player score and Register 3 stores the bit pattern representing the current state of the "moving LED."

Flowchart 2 is straightforward down to the decision block. The TMI (Test under Mask Immediate) instruction sets the Condition Code (part of the 2650 status word), and the BCFR (Branch on Condition False, Relative) returns program flow to the input statement until the Start switch is closed. A dashed line indicates that the game program can be entered at the decision block if desired.

Note that the displays will show random values until the first game is played, then the score displays will show the previous game score until the pending game is finished. The advantage of this entry point is that a simple push of a button initiates the action, in place of resetting the CPU and making a new program entry.

A similar decision block will suffice to detect the key action during the game; redefine the mask as 01 instead of 80. (These two mask values result from inputting the Key on Bit 0 and the Start switch on Bit 7 as shown in Fig. 17.) The BCTR (Branch on Condition True, Relative) instruction is shown with Condition Code set to 3, which is defined as an unconditional branch. This forces a return to the program start.

Flowchart 3 shows the two self-explanatory sequences that move the LED pattern to the left and right. Flowchart 4 details one way to perform the "end LED?" test. Register 0 is loaded from Register 3, and either 80 or 01 (depending on which point in Flowchart 1 is being tested) is subtracted from Register 0. Either instruction BRNR (Branch on Register Non-zero, Relative) or the instruction to follow is taken, depending upon the Condition Code resulting from the Subtract instruction.

If you wish to evaluate various microcomputers, build one universal set of interface hardware and program the same set of tasks with each machine. Compare the operating time and amount of memory needed. The question of "which is the best microprocessor?" suddenly takes on new meaning, and the difficulty of making a valid choice becomes apparent.

No matter which you pick as a favorite, you will soon want some form of monitor program in ROM (read only memory). Then when power is first applied to the system, it is possible for the CPU to help with the startup. The monitor program may operate a cassette recorder, simple keyboard and LED display, or it might run a Teletype or TVT. Even when you reach this point, it is likely that you will have spent less than $200 on the system exclusive of the TTY or TVT. Considering the excellent education you will have received, that is a super bargain! ■

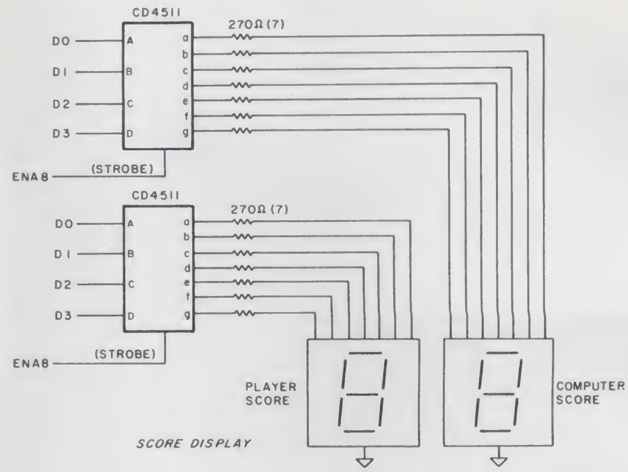
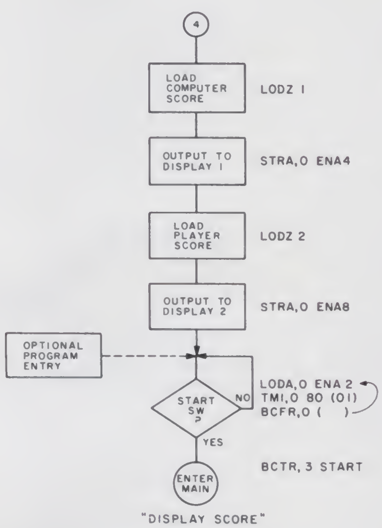
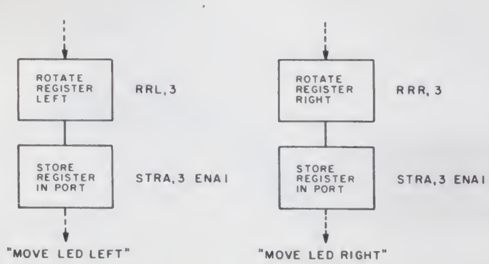


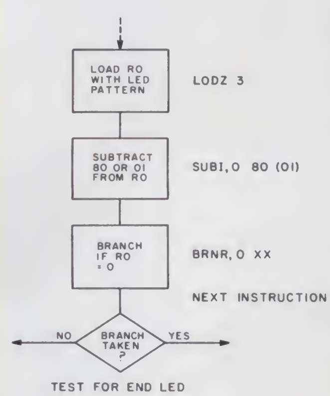
Fig. 18.



Flowchart 2.

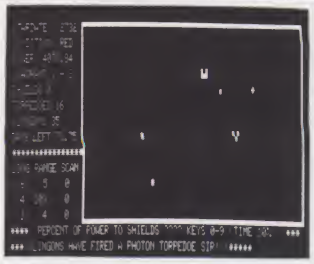


Flowchart 3.



Flowchart 4.

SOFTWARE → TRS-80 ← SOFTWARE

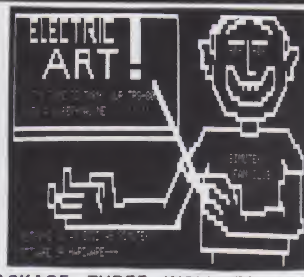


PACKAGE ONE INCLUDES: GRAPHIC-TREK "2000" — This full graphics, real time game is full of fast, exciting action! Exploding photon torpedoes and phasers fill the screen! You must actually navigate the enterprise to dock with the giant space stations as well as to avoid klingon torpedoes! Has shields, galactic memory readout, damage reports, long range sensors, etc! Has 3 levels for beginning, average, or expert players! ★ **INVASION WORG** — Time: 3099, Place: Earth's Solar System Mission: As general of Earth's forces, your job is to stop the Worg Invasion and destroy their outposts on Mars, Venus, Saturn, Neptune, etc! Earth's Forces: Androids — Space Fighters — Lazer Cannon — Neutrino Blasters! Worg Forces: Robots — Saucers — Disintegrators — Proton Destroyers! Multi level game lets you advance to a more complicated game as you get better! ★ **STAR WARS** — Manuever your space fighter deep into the nucleus of the Death Star! Drop your bomb, then escape via the only exit. This graphics game is really fun! May the Force be with you! ★ **SPACE TARGET** — Shoot at enemy Ships with your missiles. If they eject in a parachute, capture them — or if you're cruel, destroy them! Full graphics, real time game! ★ **SAUCERS** — This fast action graphics game has a time limit! Can you be the commander to win the distinguished cross! Requires split second timing to win! Watch out!

ONLY $12.95

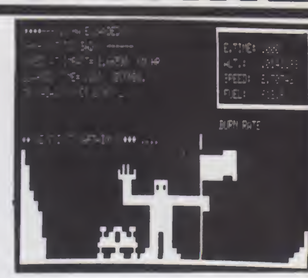
PACKAGE TWO INCLUDES: CHECKERS 2.1 — Finally! A checkers program that will challenge everyone! Expert as well as amateur! Uses 3-ply tree search to find best possible move. Picks randomly between equal moves to assure you of never having identical games. ★ **POKER FACE** — The computer uses psychology as well as logic to try and beat you at poker. Cards are displayed using TRS-80's full graphics. Computer raises, calls, and sometimes even folds! Great practice for your Saturday night poker match! (Plays 5 card draw). ★ **PSYCHIC** — Tell the computer a little about yourself and he'll predict things about you, you won't believe! A real mind bender! Great amusement for parties. ★ **TANGLE MANIA** — Try and force your opponent into an immobile position. But watch out, they're doing the same to you! This graphics game is for 2 people and has been used to end stupid arguments. (And occasionally starts them!) ★ **WORD SCRAMBLE** — This game is for two or more people. One person inputs a word to the computer while the others look away. The computer scrambles the word, then keeps track of wrong guesses.

ONLY $12.95



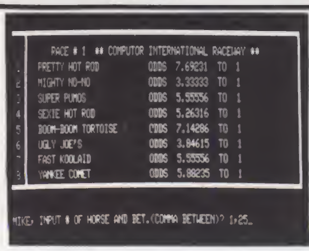
PACKAGE THREE INCLUDES: POETRY — This program lets you choose the subject as well as the mood of the poem you want. You give TRS-80 certain nouns or names, then the mood, and it does the rest! It has a 1000-word + vocabulary of nouns, verbs, adjectives and adverbs! ★ **ELECTRIC ARTIST** — Manual: draw, erase, move as well as, Auto: draw, erase and move. Uses graphics bits not bytes. Saves drawing on tape or disk! ★ **GALACTIC BATTLE** — The Swineus enemy have long range phasers but cannot travel at warp speed! You can, but only have short range phasers! Can you blitzkrieg the enemy without getting destroyed! Full graphics — real time! ★ **WORD MANIA** — Can you guess the computer's words using your human intuitive and logical abilities? You'll need to, to beat the computer! ★ **AIR COMMAND** — Battle the Kamikaze pilots. Requires split second timing. This is a FAST action arcade game.

ONLY $12.95



PACKAGE FOUR INCLUDES: LIFE — This Z-80 machine language program uses full graphics! Over 100 generations per minute make it truly animated! You make your starting pattern, the computer does the rest! Program can be stopped and changes made! Watch it grow! ★ **SPACE LANDER** — This full graphics simulator lets you pick what planet, asteroid or moon you wish to land on! Has 3 skill levels that make it fun for everyone. ★ **GREED II** — Multi-level game is fun and challenging! Beat the computer at this dice game using your knowledge of odds and luck! Computer keeps track of his winnings and yours. Quick fast action. This game is not easy! ★ **THE PHARAOH** — Rule the ancient city of Alexandria! Buy or sell land. Keep your people from revolting! Stop the rampaging rats. Requires a true political personality to become good! ★ **ROBOT HUNTER** — A group of renegade robots have escaped and are spotted in an old ghost town on Mars! Your job as "Robot Hunter" is to destroy the pirate machines before they kill any more settlers! Exciting! Challenging! Full graphics!

ONLY $12.95



PACKAGE FIVE INCLUDES: SUPER HORSERACE — Make your bets just like at the real racetrack! 8 horses race in this spectacular graphic display! Up to 9 people can play! Uses real odds but has that element of chance you see in real life! Keeps track of everyone's winnings and losses. This is one of the few computer simulations that can actually get a room of people cheering! ★ **MAZE MOUSE** — The mouse with a mind! The computer generates random mazes of whatever size you specify, then searches for a way out! The second time, he'll always go fastest route! A true display of artificial intelligence! Full graphics, mazes & mouses! ★ **AMOEBA KILLER** — You command a one man submarine that has been shrunken to the size of bacteria in this exciting graphic adventure! Injected into the president's bloodstream, your mission is to destroy the deadly amoeba infection ravaging his body! ★ **LOGIC** — This popular game is based on Mastermind but utilizes tactics that make it more exciting and challenging — has 2 levels of play to make it fun for everyone. ★ **SUBMARINER** — Shoot torpedoes at the enemy ships to get points. Fast action graphics, arcade type game is exciting and fun for everybody!

ONLY $12.95

HARDWARE → TRS-80 ← HARDWARE

MICRO SPEED

Upgrade your "slow" TRS-80 to a **SUPER FAST MACHINE!!** (2.66 MHZ) over **50% FASTER!** Some of the features:

Auto turn-off during cassette or disk access. (This means NO lost programs EVER!) (Turns back on automatically too!) MANUAL control. (Unit may be turned on or off at any time. Yes even during program execution!) Keyboard indicator light "blinks" when micro-speed is on. Stops blinking when off! **Don't wait for SARGON II or any other program!!!** Comes with easy to follow instructions. (Some soldering required.) **OR** take to your local computer store or TV-Appliance Center for quick installation. (5-10 minutes!!) Works with any model, TRS-80.

ONLY $24.95 complete

MICRO BEEP

Simple hook up: Just plug cassette remote jack into unit.

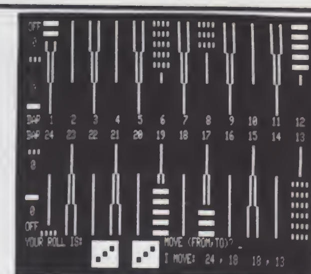
EASILY CONTROLLED FROM BASIC:

OUT 255,4 = on
OUT 255,0 = off

MICRO-BEEP make games more fun as well as provide useful sound output for professional applications!

Works with Any Model I TRS-80

ONLY $9.95 complete



PACKAGE SEVEN INCLUDES: BACKGAMMON 5.0 — 2 different skill levels make this game a challenge to average or advanced players FAST (15 second avg) Looks for best possible move to beat you! FANTASTIC GRAPHICS. Plays doubles and uses international rules ★ **SPEED READING** — Increases your reading speed. Also checks for comprehension of material. Great for teenagers and adults to improve reading skills ★ **PT 109** — Drop depth charges on moving subs. Lower depths get higher points in this fast action graphics game. ★ **YAHTZEE** — Play Yahtzee with the computer. This popular game is even more fun and challenging against a TRS-80! ★ **WALL STREET** — Can you turn your $50,000 into a million dollars? That's the object of this great game Simulates an actual stock market!

ONLY $12.95

PACKAGE SIX INCLUDES: 20 HOME FINANCIAL PROGRAMS — Figures amortization, annuities, description rates, interest tables, earned interest on savings and much, much more. These programs will get used again and again. A must for the conscientious, inflation minded person.

ONLY $12.95

SIMUTEK

✓29

Exceptional Products through Research & Imagination

Send Check, Money Order or Bank Card No. orders to:

SIMUTEK
P.O. Box 35298
Tucson, AZ 85740

Please Add 2.50
Per Order For
Postage & Handling

Master Charge

24 HOUR (7 days) HOTLINE
(602) 882-3948
(C.O.D. $3 extra)

Visa

Same Day Shipment on Bank Cards, Money Orders & C.O.D.

All Tape Programs Require a Minimum of 16K Level 2
Packages Available on Diskette (32K System) $4.25 Extra

3 or More Packages Get 10% Discount

Dealer Inquiries Invited

TRS-80 IS A REGISTERED TRADEMARK OF TANDY CORP.

Build your own microcomputer as you learn computer technology at home.

New from NRI! The First Interdisciplinary Home Study Course Ever Offered

As the microprocessor revolutionizes the computer world and microcomputers appear almost everywhere, NRI brings you a new, convenient, and effective way to keep up with this expanding technology. It's NRI's courses in Microcomputers and Microprocessors, created and designed exclusively for learning at home in your spare time.

Designed for the New Breed of Computer Technician

It's no longer enough to be just a programmer or technician. With microcomputers moving into the fabric of our lives as low-cost, easily available tools for business and home, both the programmer and technician must become total professionals. With practical knowledge of hardware, the programmer can design simpler, more effective programs. And with advanced programming skills, the technician can test and debug systems quickly and easily. The NRI course gives you simultaneous training in both skills...makes you one of this rare new breed.

Build Microcomputer, Test Instruments

NRI goes far beyond book learning to give you practical, "hands-on" experience. As you learn, you actually assemble NRI's designed-for-learning microcomputer. It performs like the finest of its kind, and features both assembly and basic language capabilities.

Every assembly step's a learning step. Using the NRI Discovery Lab® plus the NRI transistorized volt-ohm meter and CMOS digital frequency counter you also build, you perform meaningful experiments throughout your course...trace circuitry, interface components, introduce and correct problems, design your own programs, and more.

The Proven Way to Learn at Home

You don't have to worry with travel, classes, or time lost from work when you learn the NRI way. As they have for more than 60 years of teaching technical subjects, NRI brings the material to you. You study in your spare time, at your convenience, using "bite-size" lessons that program material into logical segments for easier assimilation. You perform experiments and build equipment using kits we supply. And your personal NRI instructor is always available for consultation should you have questions or problems. Over a million students have already shown the effectiveness of NRI training.

Choice of Courses

Several courses are available, depending upon your needs and background. NRI's Master Course in Microcomputers and Microprocessors starts with the fundamentals, explores basic electronics and digital theory, the total computer world, and the microcomputer. The Advanced Course, for students already versed in digital electronics, concentrates on software and the world of the microprocessor and microcomputer. In both courses, you build all instruments and your own computer.

Send for Free Catalog... No Salesman Will Call

Get the details on these exciting new courses in NRI's free, 100-page catalog. *Shows* all kits and equipment, lesson outlines, and full information, including facts on other electronics courses. Mail the coupon today and we'll rush your catalog. No salesman will ever call. Keep up with the latest technology as you learn on your own computer. If coupon has been removed, write to NRI Schools, Computer Department, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses *approved under GI Bill.*
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Amateur Radio • Basic and Advanced
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating Including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 172-040

✓Reader Service—see page 257

Lowercase for the TRS-80

It's easier than you think. Last month we ran a strictly software approach to lowercase for the TRS-80. The technique that's presented here utilizes both hardware and software.

Steven Wexler
1634 Buck Hill Drive
Huntingdon Valley PA 19006

With one memory chip, one spare OR gate and some software, you can add lowercase to the Level II TRS-80 video display in about 20 minutes.

How is this possible? The TRS-80 has a character generator that supports lowercase! Unfortunately, Radio Shack didn't include a 2102 RAM chip for the most significant ASCII bit in the video memory.

Instead, they generated a quasi-bit by inverting the second most significant bit when the character generator is enabled (the character generator is disabled for graphics). This has the effect of making ASCII control codes appear as uppercase. The ROM software video driver converts all uppercase and lowercase ASCII to control codes.

Solution

If you use a spare gate to OR the quasi-bit with a new 2102 bit, control codes will continue to be displayed as uppercase (see Fig. 1). After you load a patch to the ROM software video driver, lowercase will be displayed as lowercase. However, without the patch, lowercase will continue to be displayed as uppercase.

Words of Caution

Although this modification is relatively simple as far as modifications go, it still should not be attempted by people with little or no hardware experience.

Since any modification voids the Radio Shack 90-day warranty, wait until the warranty expires before installing lowercase. Radio Shack charges a blanket rate of $24 to repair out-of-warranty, unmodified TRS-80s. However, they charge $48 plus parts to repair modified TRS-80s. In addition, they rip out any modifications!

```
                    ORG 401EH
401E EA,7F          DEFW LW        ;Change video driver pointer
                    ORG 32746      ;Top of 16K RAM
7FEA DD,6E,03 LW    LD L, (IX + 3) ;Set HL to current cursor address
7FED DD, 66, 04     LD H, (IX + 4)
7FF0 DA,9A,04       JP C,49AH      ;Exit if not writing to screen
7FF3 AF             XOR A          ;Set reg. A to zero
7FF4 B1             OR C           ;Transfer reg. C to reg. A, setting flags
7FF5 FA,A6,04       JP M,4A6H      ;If not ASCII leave
7FF8 FE,20          CP 20H         ;Set flag for control code test
7FFA 02,7D,04       JP NC,47DH     ;Exit if control code
7FFD C3,60,04       JP 460H        ;Patch back to ROM, bypassing conversion
                                   ;to control codes
                    END
```

Listing 1. Video driver patch.

Procedure

1. Disconnect cassette, video, power and/or expansion plugs.

2. Turn the TRS-80 upside down on a non-scratch surface and remove the screws.

3. Turn the computer face up and remove the top cover. On some units you must remove the cover slowly, making sure you free up the power-on LED.

4. At this point you will see the keyboard sitting on five posts. Note that the ribbon connector connects the keyboard to the main board. The ribbon connector does not disconnect from either board. Lift the keyboard from the posts, flexing the ribbon connector as little as possible.

5. Underneath the keyboard are five spacers. Note the posts they are on and remove them.

6. Lift the main board off the posts and onto the top of the keyboard. Component side of the main board should be up.

7. Note the socket that the cable from the Level II ROM board is plugged into (Z33 or Z34). Disconnect the cable.

8. Using long-nose pliers, bend pins 11 and 12 of the new 2102 upward. Place the new 2102 on top of Z63. Pins 11 and 12 should be on the right-hand side. The new 2102 will be called Z63A.

9. Quickly and carefully solder all but pins 11 and 12 of Z63A to the respective pins of Z63.

10. Using tweezers to hold the wire-wrap posts, solder a post to Z60, pin 4, Z60, pin 5, and Z30, pin 13.

11. Using Solder Up (available at Radio Shack), desolder pins 11, 12 and 13 of Z73 from the board. Turn these pins upward with long-nose pliers.

12. Wire-wrap the following connections:

Z63A, pin 11 to Z60, pin 5
Z63A, pin 12 to Z73, pin 12
Z30, pin 13 to Z73, pin 13
Z73, pin 11 to Z60, pin 4

13. A narrow trace runs between pins 5 and 6 of Z30 out toward the left of Z30. Cut the trace with a knife.

14. Connect the Level II cable into its socket (see step 7).

15. Restore the main board into the bottom case, with the keyboard cable toward the front, component side down.

16. Restore the spacers.

17. Place the keyboard onto the appropriate posts.

18. Put the top cover in place, turn the unit over and screw together. Place the short screws

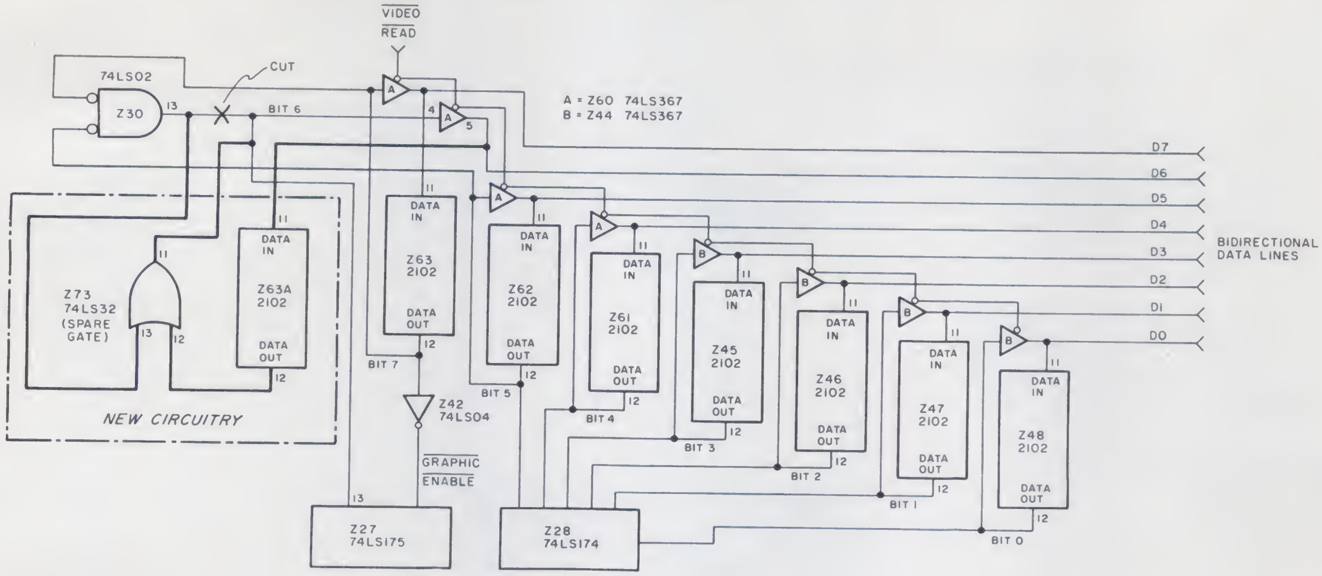


Fig. 1. Circuit modification.

on the thinnest part.

19. Reconnect the power, video, cassette and/or expansion plugs. Warning: Make certain the power plug is in the correct place; otherwise, extensive damage may result.

20. Run the following program to test the modifications.

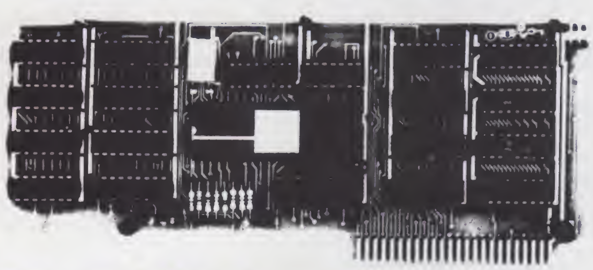
```
10 CLS
20 FOR A = 0 to 127
30 POKE 15360 + A,A
40 NEXT
50 PRINT @ 640,""
```

You should see uppercase, punctuation, numbers, uppercase again and, finally, you will see lowercase.

Software

Listing 1 contains a short patch to the ROM video driver. The listing is assembled at the top of 16K RAM.

On power-up, set memory size to 32745 and load the patch, but do not run it. Use the Break key to get the "ready" prompt. Type your name, first without using the shift key, then with the shift key. If you can see the difference, congratulations . . . you now have lowercase! ■



↙Reader Service—see page 257

Thoughts on the SWTP Computer System

Part 11 examines how to use the SWTP 32K dynamic memory board.

Peter A. Stark
PO Box 209
Mt. Kisco NY 10549

In March, we looked at how the SWTP 32K dynamic memory board works and how to change it. We will continue our examination of this board by discussing how we can use it.

If you just make it into a plain 32K board using either 24 or just 16 ICs, then you can use it the same way as any other 32K board, except that it takes less power because it has fewer ICs.

Converting to a full, one-piece 64K board is not practical; the 6800 will not support that much RAM since it also needs room for ROM and I/O.

Converting to a 64K board in two 32K chunks switched by an output port (PIA) is an intriguing idea because it opens up a whole new area of time-sharing and multiprogramming possibilities.

Converting to a 48K board for a 6800 system is the most worthwhile, but it requires much work. Normal I/O is at address 8000; if you want 48K of continuous memory, you must move this elsewhere. (In a 6809 system, however, a 48K memory board is useful, since 6809 monitors such as SWTP SBUG-E or Percom's PSYMON assume that I/O will be somewhere in high memory. Perhaps modifying to 48K would be most useful here.)

6800 System Modifications

I have modified my 6800 system to implement 48K of contiguous RAM. The memory map of my system is as follows:

0000–BFFF	48K of RAM
C000–CFFF	Percom disk controller and its ROM
D800–DFFF	Percom video board
E000–E3FF	SWTBUG in a 2708 EPROM
E400–F7FF	Other routines in 2708 EPROMs
F800–FBFF	I/O ports 0 through 7
FC00–FFF7	2708 EPROM, temporarily empty
FFF8–FFFF	SWTBUG's reset and interrupt vectors

Note the heavy use of 2708 EPROMs. Although I have the MP-A2 CPU card and several 2716 EPROMs that could be plugged into it, I have a separate 2708 EPROM board with a number of 2708s. There are several reasons for this.

First of all, the 2708 is less expensive and more available than the 2716. But this is outweighed by the need to build a 2708 board.

More important, although I am still using SWTBUG (I had to burn it into an EPROM to change all I/O references from 8000 to F8), I am planning to switch over to a modular "monitor to end all monitors," which I mentioned in the February 1980 installment. Although it will be spread out over more than one EPROM, each EPROM will stand on its own, and future improvements will be made by just burning one new EPROM at a time. Smaller chunks of program will have to be updated with the 1K 2708 than with the 2K 2716.

But there is a more important reason for the use of the 2708. I wanted the whole region from E000 through FFFF to be EPROM, except for a small region used up by I/O. For the sake of simplicity, I let the EPROM board address-decoding circuitry handle I/O address decoding too. The I/O on the motherboard simply replaces one EPROM. With the 2708, I/O only uses up 1K of memory; with a 2716 it uses up 2K.

Making this change, I had to proceed in a carefully planned sequence to prevent boxing myself into a corner. Follow these steps:

1. Change all I/O references in SWTBUG from 80xx to F8xx and burn it into 2708 EPROMs by changing each 80 to an F8 in SWTBUG locations E10A, E156, E290, E2AE, E2B4 and E2C7. I then burned SWTBUG into two 2708s; one got everything in addresses E000 through E3F7, while the other got the reset and interrupt vectors given in the SWTBUG listing as locations E3F8 through E3FF. This latter 2708 will eventually be addressed from FC00 through FFFF, and these vectors will appear in FFF8 through FFFF, rather than the addresses listed in the program listing. It would have been possible to burn SWTBUG into just one 2708 and then modify the EPROM board address decoding to put it into two address locations at the same time, but it is much simpler to just use two EPROMs instead. Once the 2708s were burned, I put them aside while I continued.

2. Modify the disk operating system and store it on cassette. In SWTP and SSB disk systems, the disk operating system (DOS) is stored on disk. If you don't modify the DOS before I/O addresses are changed, you will not be able to modify it later because you will have no way of reading the disk. Hence, any changes will have to be done first and stored on some other medium, such as a cassette.

In the case of the Percom DOS, no changes are required since it does not refer to any I/O addresses (its controller is not plugged into the I/O bus).

The SSB disk controller has a bootstrap program, some routines in ROM and the rest of the DOS on the disk. You will have to change all I/O addresses in both from 80xx to F8xx, burn a new ROM and store the disk-resident portion of the DOS on cassette.

In the case of the SWTP MF-68 disk, the disk bootstrap in SWTBUG has already been changed as part of the above, so only the disk-resident DOS must be changed, and again stored on cassette so it can be read back in later. In mini-Flex, the locations shown in Example 1 must be changed from 80 to F8. Then save locations 7080 through 7FFF with a starting address of 7100. In Flex 2.0, the locations in Example 2 must be

72E3	72E9	7F1E	7F26	7F34	7F43	7F4B	7F50	7F58	7F62
7F6E	7F76	7F86	7F97	7FAA	7FB5	7FD9	7FDE	7FE8	

Example 1.

A720	A776	AF81	AF87	BEA7	BEA9	BEC0	BEC8	BED6	BEEC
BEF4	BEF9	BF01	BF0B	BF1D	BF25	BF35	BF4D	BF61	BF7F
BF84	BF8E								

Example 2.

changed from 80 to F8.

In FLEX disk systems, there is one big problem: once you move your I/O out of 8000, you will not be able to boot any of your old disks. Using the DOS you have modified and just saved (on cassette, for instance), you will be able to bring up FLEX by loading it from cassette and then starting it at either 7100 or AD00, depending on the version. Using a modified NEWDISK (see the section on patches), you can initialize new disks and put this DOS on them. These will then boot with the D command in your modified SWTBUG. But disks NEWDISKed earlier will not boot, even if you replace the DOS on them with the new version.

This can be a major problem if you do not have a FLEX disk system now, but plan to get one after you move your I/O. You will have no way of reading in the DOS to make the required changes, unless you have a friend with a disk system who can make these changes for you and give you a cassette of DOS.

I suspect that a similar problem exists with the SSB disk; the Percom disk does not have the problem. You may also have a lot of trouble if you go to a standard floppy or hard disk in the future.

3. Modify the ROM board to delete the region used up by the I/O ports from the ROM memory area. Most ROM boards take up multiples of 4K of addresses. Hence, if we address part of the board at the SWTBUG area of E000–E3FF, the ROM board will take up the full 4K from E000 through EFFF, whether we like it or not. Likewise, since we need the reset and interrupt vectors up at FFF8 through FFFF, the ROM board will take up the full area from F000 through FFFF . . . again, whether we use the rest or not. This interferes with our putting the I/O addresses up into high memory. So we must wire up the ROM board to leave some memory unassigned to the board, and therefore available for I/O.

How you do this depends on the ROM board. The Micro Works 2708 EPROM board, for example, is already designed for just this with a jumper that can be used to delete any 1K segment from the board so it can be used for I/O. In fact, this board even has the circuitry for decoding this I/O address range and feeding the I/O address decoders on the motherboard. In that sense, the Micro Works EPROM board is a perfect candidate for this job.

My system contains the 16K 2708 EPROM board available from Walter Wimberly (2914 Sunrise Drive, Orlando FL 32803) for $27.50 for a bare board. This board occupies the 8K slot from E000 through FFFF, except that a 1K chunk has been deleted at F800–FBFF by making the change shown in Fig. 1.

In Fig. 1a the circuit before the change had U21B generating a Board Select signal, which enabled the data output buffers when any ROM was selected; U25B was a decoder that enabled one of four EPROMs . . . in this case, U17 (although any of the decoder outputs could have been used).

After the change, pins 12 and 13 of U21B are freed up and connected to the decoder instead. Now, whenever the address corresponding to U17 is selected, a low signal to U21B turns off the Board Select signal, thereby disabling the board outputs. (This same signal is also used for I/O decoding on the motherboard.)

4. Change the motherboard I/O decoding from 80xx to F8xx. IC6 on the motherboard has an output on pin 11 that goes low when an address in the range of 8000 through 8FFF is selected. I could have rewired this IC, but it was much simpler to use the decoding already on the EPROM board to turn on I/O at the exact same time as it turned off the EPROM output data. This is exactly the same idea implemented on the Micro Works EPROM board, though in a different way.

In my case, I simply soldered a two-inch length of wire to U25B, pin 10, shown in Fig. 1b, and connected the other end to pin 11 of an IC header. A header is a small plug designed to fit into an IC socket. It's normally used either on the end of a cable to connect different boards together or to mount resistors or other small parts so they can be plugged into an IC socket.

I then plugged the EPROM board into the last 50-pin connector on the motherboard, just in front of the I/O decoder/driver ICs mounted on the motherboard. I unplugged IC6 from the motherboard and plugged in the header instead of the IC. The two-inch length of thin wire was a comfortable fit. This arrangement provides perfect I/O decoding without my even having to pull the motherboard out of the cabinet for changes, and if I ever want to go back to I/O at 8000, I just unplug the header and plug IC6 back in. At this point, I plugged in the dynamic RAM board and promptly had a 48K system operating.

Patches

It was then necessary to patch software to make it run properly. One problem is that much software—such as BASIC—monitors the ACIA on port 1 to look for a break or control C. If I simply entered 00 into locations 8004 and 8005 before running these programs, I could make sure that the software would never get a control C, and so get it to run. Nevertheless, there was still much patching required. The following is a list of locations that need to be patched.

Percom Super BASIC, version 1.09. Change 80 to F8 in locations 0584, 058A, 059A, 05A1, 05AC, 05B1 and 1C0C to move the I/O. Then change location 0150 from 8000 to A000 to allow BASIC to use the full 40K of memory up to 9FFF.

Mini-FLEX NEWDISK command. Change the following locations from 80 to F8: 0466, 046C, 047B, 0487, 048C, 0492, 054C, 0556, 0563, 0579, 057E, 0585, 058E, 0596, 05B8.

Percom "Touchup" version of the TSC Text Editor. Change locations 15F0, 1CF7, 1CFA and 1D02 from 80 to F8. Also change location 16E6 from 80 to A0 so the editor will use the full 40K for its files.

TSC Debug package. Change locations 410F from 80 to F8.

Percom assembler. Change 80 to F8 in locations 010F, 02F8, 0331 and 0334.

Cores editor/assembler. Change 80 to F8 in locations 0296, 029B, 029E, 1680, 17A3, 1A16, 1A60, 1A70 and 1A87.

TSC Text Processor. Change

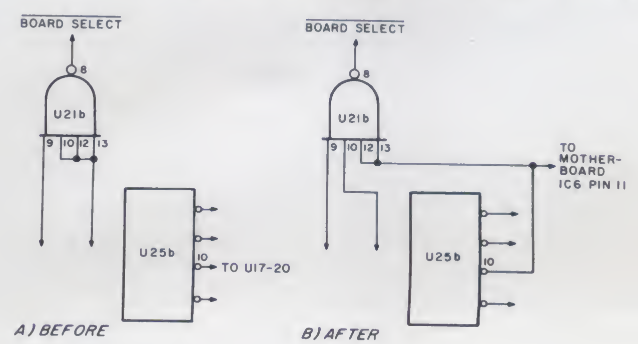


Fig. 1. 16K EPROM board modification.

locations 1472 and either 1478 or 1479, depending on the version, from 80 to F8.

SWTP EPROM programmer program—either the original 2716 version or the modified 2708 version I described in my February article. Change 80 to F8 in locations 00F5, 053D, 0540 and 054B.

There are obviously other programs that also need patching, but you get the point . . . this conversion is far from simple. I converted my 32K board to 48K mostly to see if it could be done, but I can't in all honesty recommend it for use with the 6800. On the other hand, converting to 48K for use with the 6809 is an excellent move.

> **"I converted my 32K board to 48K to see if it could be done, but I can't recommend it for use with the 6800 . . . converting to 48K for use with the 6809 is an excellent move."**

Motherboard Operation

Let's return to the SWTP motherboard. There are actually three SWTP motherboards now—the old MP-B 6800 motherboard, the newer MP-B2 board and the brand new 6809 board.

I must apologize for a mistake in the very first installment of this series (March 1979). I described the differences between the older SWTP MP-B motherboard and the newer MP-B2 motherboard, but gave the wrong information for updating the older board.

With all SWTP 6800 systems, I/O ports are located starting at address 8000. For instance, port 0 is 8000–8003, port 1 is 8004–8007, and port 7 is 801C–801F. Addresses 8020–803F are reserved for ports 8 through 15, but to get these additional ports you have to add a second motherboard to provide the additional decoding and the additional connectors.

The whole area above 803F—that is, from 8040 and up—is unused by I/O. However, to save some logic, the designer of the motherboard did not include full decoding of the I/O addresses. (For a more comprehensive discussion of address decoding, see Kilobaud Klassroom No. 11, August 1978.) Fig. 2 shows a partial diagram of the decoding circuits for the MP-B motherboard.

IC3 and IC6 are 74S138 decoders. In order for the decoder to operate, the CS (chip select) inputs must be high, and the two $\overline{CS}$ (not chip select) inputs must be low. Once this requirement is satisfied, then the IC will look at its C, B and A inputs and decode the binary number on them into one of eight outputs. Depending on the binary number on the inputs, it will ground one of the outputs and make the others high.

IC6 actually has ten outputs, but only one of them is used; this is the same pin 11 that I am using in my conversion to a 48K system. When IC6 is used, in order for this output to be grounded, the input to the CBA pins must be the binary number 100, which is 4. So we see that this IC will provide an output whenever bit A15 is 1, A14 and A13 are both 0, and A5 is also 0 or low. In this case, A15 is the leftmost, or most significant, bit of the address, while A0 is the rightmost bit.

Using the symbol x for a bit that can be either 0 or 1—often called a "don't care"—we see that IC6 will provide a grounded or low output for addresses that look like this:

100x xxxx xx0x xxxx

Since the hexadecimal digits 8 and 9 both start with 100 (either 1000 or 1001), the I/O ports will be decoded as addresses in the 8000 and 9000 range, as long as bit A5 is a zero. (A5 would be 1 for addresses 8020–803F, so requiring it to be 0 guarantees that the circuitry on this motherboard doesn't respond to the addresses that were set aside for ports 8 through 15.)

Once IC6 sends out a low output to indicate that we are in the right range, IC3 separates specific addresses for each of the eight ports by looking at bits A4, A3 and A2. For instance, the first address for port 7 would be 801C, or

1000 0000 0001 1100

The 111 grouping represents bits A4, A3 and A2. Since they are all 111, or 7, output 7 from IC3 is grounded. This is an enable signal for port 7.

As mentioned before, the I/O ports will respond to any address that starts with the bits 100, meaning that they will take up all the addresses from 8000 up through 9FFF. Thus, 32 addresses for I/O will use up a full 8K of address space, making it useless for any other purpose.

This is where we started our discussion of addresses in the first installment of this series (March 1979). I wrote that with a simple change we could require bit A12 to be 0, and thus force the motherboard to respond only to addresses starting with 8. This would free up the entire range from 9000–9FFF for other use, such as adding another 4K memory board.

My suggestion had been to break the ground going to pin 5 of IC3, the $\overline{CS}$ pin, and connect that pin instead to A12. Now IC3 would select an I/O address only when A12 was low, and we would be done. As far as IC3 was concerned, I was right. But I forgot about the gates getting their signal directly from the output of IC6.

The motherboard has a set of data bus buffers that connect the main data bus—the CPU bus—running on the front part of the motherboard to the data bus going along the back of the board to the I/O boards. Since data has to go both ways between these two, the buffering is done with bidirectional transceivers.

The function of the two gates and inverter is to control which way these transceivers transmit. Specifically, if the output from IC6 is low and the Read/Write signal indicates a read, these gates will turn the transceivers around and send data from the I/O ports back to the CPU. Thus, whenever we read (load) from addresses starting with 8 or 9, we will get data coming from the I/O data bus back to the CPU data bus. This happens even if we connect A12 to IC3. So if we make this change and then install a 4K memory board in addresses 9000–9FFF, every

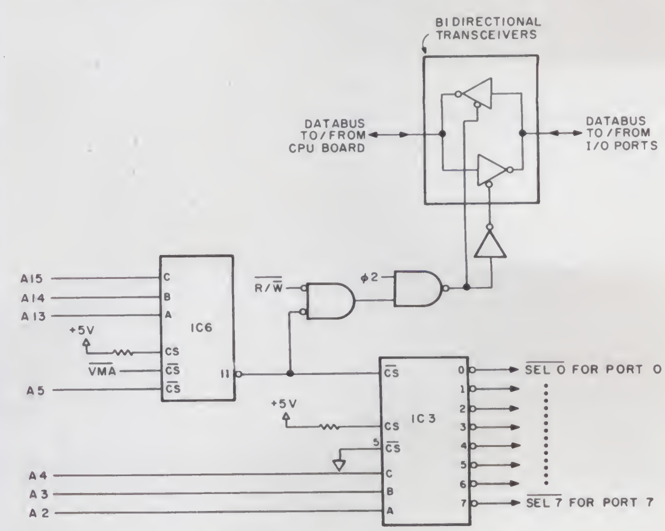


Fig. 2. MP-B address decoding.

time we try to read from this board we will get not only the memory board's data, but also some garbage data from the I/O data bus.

To solve the problem, we have to disable IC6 during all addresses of 9000 or above and not bother changing IC3 at all. We could use the CS pin; instead of keeping it high by connecting it to +5 volts through a resistor, we could connect it to A12 through an inverter so it would go high only when A12 was low. Since IC4 and IC5 have some unused gates, one could be used for the job (except that the unused gates have their inputs connected on the motherboard, and so some traces have to be cut to free them up).

Alternatively, if we added a little more logic, we could improve the decoding even more. Fig. 3 shows how adding a 7425 dual four-input NOR will allow decoding all the address lines from A12 down through A5. (Don't get fooled by the 7425 symbols in Fig. 3—this is just an alternative symbol for NOR, which indicates that the output is high when all of the inputs are low.)

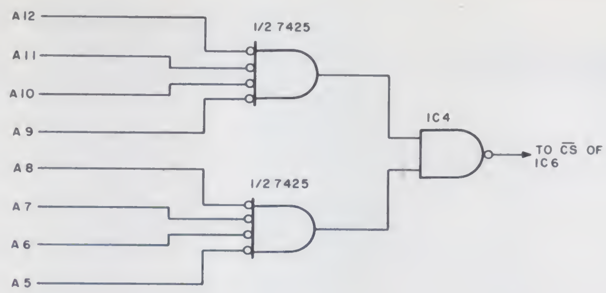


Fig. 3. Providing full decoding for the MP-B.

When all address lines from A12 through A5 are low, the two inputs into the NAND gate (one of the unused gates in IC4) are both high, which makes the output low. Simply connect this to the CS input of IC6, instead of the A5 signal it now has. With this decoding, all the addresses above 801F will be opened up. Hence, you could now add other devices, or even memory, up there.

We might question the practicality of this. We need low memory, not high memory. That is, even if you put more memory at addresses above 801C, most programs such as BASIC will not be able to use it (unless you PEEK and POKE data up there).

To really make this useful, you have to move I/O out of the 8000 range, as described above. The problem is that it makes your system nonstandard. Every piece of software you get from now on will have to be carefully examined to make sure it doesn't conflict with your new address assignments. And unfortunately, this will bring you down to the level of all the S-100 users, who have that problem all the time. It kills the one feature of our 6800 that makes it so easy to use: almost all 6800 systems are similar in their memory layout, so you can buy a new piece of hardware or software, plug it in, and it works.

The Missing Sector Hole

If you have the MF-68 (or PTA) disk system, try this little experiment. Boot the system, take the disk out and then cover with a piece of electrical tape the small hole on the disk which is used to allow the drive to sense the position of the sector hole. (Put a small piece of paper over it first so you don't touch the sticky part of the tape to the disk and stop it from turning.) Now see if it still works.

As you will see, it does. The controller uses the sector holes only during booting of the system, or when initializing a new disk. During routine reading and writing, the sector hole is not used.

Now, how do we make use of this little tidbit? In the second installment of this series (June 1979), I described how to punch an extra hole in the diskette jacket so you could write on the back of the disk as well as on the front. I mentioned that this was only necessary in those disk drives that did not have dual sector hole sensors. The Shugart SA400 does not; many other manufacturers' drives, though not all, do.

Thus, the early MF-68 systems, which used the Shugart drive, can't use the back of the disk without that second hole, while the newer systems, which use other drives, can.

If you have the Shugart drives, however, all is not lost. Since the sector hole is not used in routine reading and writing, the only reason you need this extra hole is to initialize the back of the disk or boot it. But if you can initialize it on a dual sensor drive, then you can continue to use the back of it on your Shugart drive even if it does not have the second hole.

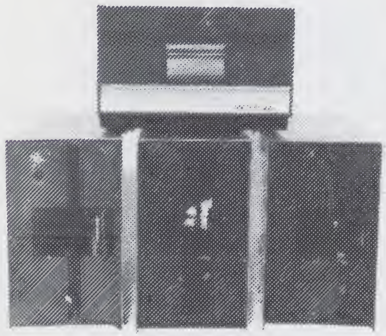
ROM Monitors Coming

Next time, we will review some of the ROM monitors—including SWTBUG, SMARTBUG, MX-68RT, JOEBUG and GMX-BUG—available for the SWTP system. We will try to come up with the "monitor to end all monitors." ■

save on add-on products for........ TRS-80

TRS-80 COMPATABLE DISK DRIVES

The largest family of disk drives from the largest supplier, drives come complete with power supply and cabinet



TF-1 Pertec FD200, 40 track	$389
TF-3 Shugart SA400, same as tandy	$389
TF-5 MPI, B51 40 track	$379
TDF-1 Dual Head, 35 track	$499
MTI-40 Drive, 35 & 40 track	**$369**

220 versions available

NOW! SUPER & MAX DISKS

SUPER: 77 tracks on one 5¼ diskette that's 197K of storage with New DOS+ $695

MAX DISK II: 9.8 Megabyte on one disk using Winchester technology. Unit is sealed in unit for added protection $5349

LINE PRINTERS

NEW!

BASE 2 ONLY $599.

The new base 2 printer, prints 60 lpm, 80 & 132 col. formats. It has tractor feed w/RS232, IEEE & centronic interface.

DP800 ANADEX 80 column, 112 cps	$949
LP779 Centronics 779	$1099
LP730 Centronics 730	$950
LP700 Centronics 700	$1395
LP701 Centronics 701	$1759
LP702 Centronics 702	$1895
Spin-1 Spinwriter	$2525

MOD II DISK DRIVES NOW AVAILABLE

ADD-ON DRIVE FOR ANY MICROCOMPUTER

DOES NOT INCLUDE POWER OR CABINET

Pertec FD200	$282
Pertec FD250 (dual head)	$349
Shugart SA400 (unused)	$286
Shugart SA800	$479
MPI B51	$279
MPI B52 (dual head)	$349

SOFTWARE

NEW DOS+ over 200 modifications, corrections and enhancements to **TRS DOS** Includes utilities.

35 Track NEWDOS+	$99
40 track NEWDOS+	$110
AJA Word Processor	$75
AJA Business Program	$250
Racet Infinite Basic	$49.95
Disk Drive Alignment Program	$109
Radex Data Base	$99.95
Electric Pencil	$150

NEW PRODUCTS

Small Systems Interface (RS232)	$49.95
Expansion Interface W32K	$499.95
AC Line Filter	$18.95
AC Isolater (6 connecters)	$47.95
Modem	$179.95
Verbatim 5¼ Diskettes	$3.39
16 Key Pad Kit (no soldering)	$68
16K Memory (Keyboard)	$89
16K Expansion (Interface)	$86

ALL PRICES CASH DISCOUNTED • FREIGHT FOB/FACTORY

MICROCOMPUTER TECHNOLOGY INCORPORATED ✓30

3304 West MacArthur
Santa Ana, CA 92704
(714) 979-9923

VISA · master charge THE INTERBANK CARD

Apparat, Inc.

7310 E. Princeton Ave.
Denver, CO 80222
(303) 758-7275

Accurate Voltage Dividers

Using your computer, you can design accurate voltage dividers easily and quickly.

Robert H. Penoyer
123 N. New, Apt. D
Monterey Park CA 91754

Voltage dividers are used for a variety of reasons. Most everyone who works with electronic circuits has encountered the simple voltage divider composed of a pair of resistors. Such a divider is shown schematically in Fig. 1.

You can easily determine the node voltage, V_n, if you know the source voltage V_s. The following equation determines V_n (see Fig. 1).

$$V_n = \frac{V_S R_2}{R_1 + R_2}$$

The Problem

The situation becomes much more difficult if you have to design your own voltage divider. For example, let's say you want to scale down the 300 volt output of a power supply to a safe 5 volts for use at a test point or as an input to an A/D converter for input to your computer. This can be done by using the equation in Example 1.

$$R_1 = \left[\frac{V_S - V_n}{V_n}\right] R_2.$$

By letting $V_S = 300$ and $V_n = 5$, we find that

$$R_1 = \left[\frac{300-5}{5}\right] R_2$$

or

$$R_1 = 59\ R_2.$$

Example 1.

This equation determines that the *ratio* R_1/R_2 is 59. But what standard pair of resistors will give you what you need? Ah, there's the rub!

The Solution

Let's say you are willing to use 2 percent tolerance resistors for the sake of accuracy. The most direct way to find the ratio needed is to divide all 2 percent resistor values by each other until the ratio nearest the required ratio is found. This is tedious, but what better job for your computer!

Using the program listing, you can determine that $R_1 = 3300$ Ohms and $R_2 = 56$ Ohms. These resistance values will yield an unloaded node voltage of about 5.006 volts. (That's an error of only 0.12 percent.) The computer doesn't know how much power you are willing to dissipate, so you will probably want to multiply the value of each of these resistors by 100. Their relative ratio won't be affected.

Fig. 1. A simple voltage divider.

The Program

The program operates in one of two modes. In one mode you provide the computer with V_s and V_n (refer to Fig. 1), and it finds the resistor values you need along with the true node voltage and the resulting voltage error in percent. In the other mode, you provide the desired ratio, R1/R2, and the computer attempts to match this ratio. A list of variables used in the program is given in Table 1.

A(I), A(J)	Elements of one decade of standard resistor values
C	Comparison index
D	Resultant node voltage
E$	String variable for "Next Case"
I, J	Resistor pair indices
K	Best comparison index thus far
L, M	Resistor pair indices of best resistor pair thus far
N	Desired node voltage
P	Ratio magnitude multiplier
R	Desired or required ratio
R3	Mantissa of $\log_{10}$(R)
R4	Integer value of $\log_{10}$(R)
S	Source voltage
T	Upper loop limit 24 for 2%, 12 for 5%, 48 for 1%
X	Mode 1 = Voltage, 2 = Ratio

Table 1. Variables used.

The program uses logarithms to compare the required ratio R (see Table 1) with the ratio currently under consideration, A(I)/A(J). Recall that $\log_{10}(n)$ of a decimal number contains a fractional part, or mantissa, and a whole number part, or characteristic. The mantissa completely defines n except for its order of magnitude, which is determined by the characteristic. The program ignores the order of magnitude of all ratios until the best resistor pair has been selected. Thus, only the mantissa is used.

```
110  DATA 12
120  DATA 10, 12, 15, 18
130  DATA 22, 27, 33, 39
140  DATA 47, 56, 68, 82
```

Fig. 2. Substitute these lines if you want to use 5 percent resistors in your voltage divider designs.

The strategy used in the program is to find $\log_{10}$(R) using R3 = LOG(R)/LOG(10) and then to retain only the mantissa of R3 using R3 = ABS(R3 – INT(R3))

Program listing.

```
10 PRINT"                DIVIDER"
20 PRINT
30 PRINT"FINDS VOLTAGE DIVIDER RESISTOR VALUES"
40 PRINT"USING 2% RESISTORS.  PROGRAM CAN USE"
50 PRINT"EITHER A KNOWN RATIO OF ONE RESISTOR TO"
60 PRINT"ANOTHER OR IT CAN USE A GIVEN SOURCE"
70 PRINT"VOLTAGE AND DESIRED UNLOADED NODE"
80 PRINT"VOLTAGE."
89 REM---DIMENSION FOR 1,2,OR 5% VALUES
90 DIM A(50)
100 RESTORE
109 REM---ESTABLISH UPPER LIMIT FOR LOOPS
110 DATA 24
119 REM---ONE DECADE OF RESISTOR VALUES
120 DATA 10,11,12,13,15,16,18,20
130 DATA 22,24,27,30,33,36,39,43
140 DATA 47,51,56,62,68,75,82,91
149 REM---READ UPPER LOOP LIMIT
150 READ T
159 REM---READ RESISTORS INTO ARRAY
160 FOR I=1 TO T
170 READ A(I)
180 NEXT I
190 PRINT
200 PRINT
209 REM---DETERMINE PROGRAM MODE
210 PRINT"VOLTAGE DIVIDER (1) OR KNOWN RATIO (2)";
220 INPUT X
230 IF X=2 THEN 280
240 IF X=1 THEN 800
249 REM---INPUT VALUE ERROR
250 PRINT
260 PRINT"YOU MUST INPUT EITHER '1' OR '2'"
270 GOTO 190
280 PRINT
290 PRINT
300 PRINT"WHAT IS DESIRED RATIO";
310 INPUT R
320 GOTO 350
330 PRINT
340 PRINT"REQUIRED RATIO IS";R
349 REM---PRESET COMPARISON INDEX
350 K=1E38
359 REM---FIND LOG BASE 10 OF R
360 R3=LOG(R)/LOG(10)
369 REM---USE ONLY MANTISSA OF R3
370 R3=ABS(R3-INT(R3))
373 IF R3<=.99 THEN 380
375 R3=0
377 R=10*R
379 REM---COMPARE ALL POSSIBLE RATIOS
380 FOR I=1 TO T
390 FOR J=1 TO T
399 REM---GET RATIO
400 C=A(I)/A(J)
409 REM---FIND LOG BASE 10 OF C
410 C=LOG(C)/LOG(10)
419 REM---USE ONLY MANTISSA OF C
420 C=ABS(C-INT(C))
429 REM---COMPARE MANTISSAS
430 C=ABS(R3-C)
438 REM---IS C SMALLER THAN INDEX K ?
439 REM---IF NOT THEN GO TO NEXT RATIO
440 IF C>K THEN 520
449 REM---IS C EXACT ?
450 IF C<>0 THEN 490
459 REM---C IS EXACT. SAVE INDEXES AND EXIT LOOP
460 L=I
470 M=J
480 GOTO 540
489 REM---C SMALLER THAN K. SAVE ALL INDICES
490 K=C
500 L=I
510 M=J
520 NEXT J
530 NEXT I
539 REM---PRESET POWER OF 10 MULTIPLIER
540 P=1
549 REM---FIND LOG BASE 10 OF R
550 R4=INT(LOG(R)/LOG(10))
559 REM---BRANCH IF R AND A(L)/A(M) ARE SAME ORDER OF MAGNITUDE
560 IF R4=INT(LOG(A(L)*P/A(M))/LOG(10)) THEN 600
569 REM---MAGNITUDES MUST BE CORRECTED
570 IF R4>INT(LOG(A(L)*P/A(M))/LOG(10)) THEN P=P*10
580 IF R4<INT(LOG(A(L)*P/A(M))/LOG(10)) THEN P=P/10
589 REM---RETEST MAGNITUDES
590 GOTO 560
600 PRINT
610 PRINT"BEST AVAILABLE RATIO IS";A(L)*P)/A(M)
620 PRINT
629 REM---BRANCH IF IN VOLTAGE MODE
630 IF X=1 THEN 930
639 REM---ENSURE BOTH RESISTORS   10
640 IF A(L)*P>=10 THEN 690
649 REM---RESISTORS TOO SMALL
650 A(L)=A(L)*10
660 A(M)=A(M)*10
669 REM---RETEST RESISTOR VALUES
670 GOTO 640
689 REM---PRINT BEST RESISTOR VALUES
690 PRINT
700 PRINT"USING R1 =";A(L)*P;"AND R2 =";A(M)
710 PRINT
720 PRINT
730 PRINT"NEW CASE (Y,N)";
740 INPUT E$
750 IF E$="Y" THEN 100
760 IF E$="N" THEN END
770 PRINT
779 REM---INPUT VALUE ERROR
780 PRINT"YOU MUST INPUT 'Y' OR 'N'"
790 GOTO 710
799 REM---VOLTAGE MODE SELECTED
800 PRINT
810 PRINT
820 PRINT"WHAT IS SOURCE VOLTAGE";
830 INPUT S
840 PRINT"WHAT IS DESIRED NODE VOLTAGE";
850 INPUT N
860 IF N<S THEN 900
869 REM---INPUT VALUE ERROR
870 PRINT
880 PRINT"NODE MUST BE LESS THAN SOURCE"
890 GOTO 800
899 REM---FIND REQUIRED RATIO
900 R=(S-N)/N
910 GOTO 330
929 REM---FIND RESULTING NODE VOLTAGE
930 D=S*A(M)/(A(M)+(A(L)*P))
940 PRINT"NODE VOLTAGE IS";D
950 PRINT"VOLTAGE ERROR IS ";100*(D-N)/N;"PERCENT"
960 GOTO 640
```

so that if R = 123, then R3 = 0.089905111. If R = 12.3, then R3 = 0.089905111. If R = 0.123, then R3 = 0.089905111.

The result of this approach is that only one decade of standard resistor values is needed. For example, if a ratio of 325 is required then R3 = 0.511883361.

During a complete search of 2 percent resistor values, the ratio 39/12 is evaluated. It happens that 39 Ohms divided by 12 Ohms is 3.25. For 3.25, R3 = 0.511883361. Therefore, except for the order of magnitude, the combination of 39 Ohms and 12 Ohms is exactly what is needed. Having found the ideal resistor pair, the program then corrects for magnitude and yields an answer of 3900 Ohms and 1200 Ohms.

The remarks provided in the listing make the program self-explanatory, but a few more comments are in order. The program requires that realistic resistor values be generated. For this reason, it will always yield resistor values of 10 Ohms or more. However, as mentioned earlier, the specific values generated may not suit your needs.

Instead of 3300 Ohms and 56 Ohms as in the first example, you may want to use 330000 Ohms and 5600 Ohms or 330 Ohms and 5.6 Ohms. Simply put, by multiplying or dividing each of the computer-generated resistor values by 10 as many times as needed, you will always end up with standard resistor values.

Sometimes more than one set of resistor values will do the same job. If, for example, you needed a divider that would divide the source voltage, V_s, in

```
                DIVIDER

FINDS VOLTAGE DIVIDER RESISTOR VALUES
USING 2% RESISTORS.  PROGRAM CAN USE
EITHER A KNOWN RATIO OF ONE RESISTOR TO
ANOTHER OR IT CAN USE A GIVEN SOURCE
VOLTAGE AND DESIRED UNLOADED NODE
VOLTAGE.


VOLTAGE DIVIDER (1) OR KNOWN RATIO (2)? 1


WHAT IS SOURCE VOLTAGE? 300
WHAT IS DESIRED NODE VOLTAGE? 5

REQUIRED RATIO IS 59

BEST AVAILABLE RATIO IS 58.9285715

NODE VOLTAGE IS 5.00595948
VOLTAGE ERROR IS  .119189508 PERCENT

USING R1 = 3300 AND R2 = 56


NEW CASE (Y,N)? Y


VOLTAGE DIVIDER (1) OR KNOWN RATIO (2)? 2


WHAT IS DESIRED RATIO? 47.36

BEST AVAILABLE RATIO IS 47.2527473


USING R1 = 4300 AND R2 = 91


NEW CASE (Y,N)? N
```

Sample run.

half, then any resistor value would work if R1 = R2. The computer will only provide you with one answer. It will be valid but it won't be general.

Under worst-case conditions my PET will find a solution in about 50 seconds. If an exact answer is possible, the execution time can be much shorter.

Figs. 2 and 3 show how to substitute 5 percent and 1 percent standard resistor values into the program in place of the standard 2 percent values provided in the program listing. You should be aware of several problems if you attempt this.

First, 5 percent values may be too sloppy. Second, there are 24 standard 2 percent values requiring 576 comparisons and 50 seconds execution time. There are 48 standard 1 percent values requiring 2304 comparisons. Thus, use of 1 percent values will require more than 3 minutes (worst case) to find a solution. Your specific needs will determine which type of resistors to use.

In any case, by using this program you will reap two benefits: you will get the best possible answer; you won't have to work very hard. ■

```
110  DATA 48
120  DATA 10,10.5,11,11.5,12.1,12.7,13.3,14,14.7,15.4,16.2,16.9,17.8,18.7,19.6,20.5
130  DATA 21.5,22.6,23.7,24.9,26.1,27.4,28.7,30.1,31.6,33.2,34.8,36.5,38.3,40.2,42.2,44.2
140  DATA 46.4,48.7,51.1,53.6,56.2,59,61.9,64.9,68.1,71.5,75,78.7,82.5,86.6,90.9,95.3
```

Fig. 3. Substitute these lines if you want to use 1 percent resistors in your voltage divider designs.

NEW! TPM* for TRS-80 Model II
NEW! System/6 Package
Computer Design Labs

Z80* Disk Software

We have acquired the rights to all TDL software (& hardware). TDL software has long had the reputation of being the best in the industry. Computer Design Labs will continue to maintain, evolve and add to this superior line of quality software.
— Carl Galletti and Roger Amidon, owners.

Software with Manual/Manual Alone

All of the software below is available on any of the following media for operation with a Z80 CPU using the CP/M* or similar type disk operating system (such as our own TPM*).

for TRS-80* CP/M (Model I or II)
for 8" CP/M (soft sectored single density)
for 5¼" CP/M (soft sectored single density)
for 5¼" North Star CP/M (single density)
for 5¼" North Star CP/M (double density)

BASIC I

A powerful and fast Z80 Basic interpreter with EDIT, RENUMBER, TRACE, PRINT USING, assembly language subroutine CALL, LOADGO for "chaining", COPY to move text, EXCHANGE, KILL, LINE INPUT, error intercept, sequential file handling in both ASCII and binary formats, and much, much more. It runs in a little over 12 K. An excellent choice for games since the precision was limited to 7 digits in order to make it one of the fastest around. $49.95/$15.

BASIC II

Basic I but with 12 digit precision to make its power available to the business world with only a slight sacrifice in speed. Still runs faster than most other Basics (even those with much less precision). $99.95/$15.

BUSINESS BASIC

The most powerful Basic for business applications. It adds to Basic II with random or sequential disk files in either fixed or variable record lengths, simultaneous access to multiple disk files, PRIVACY command to prohibit user access to source code, global editing, added math functions, and disk file maintenance capability without leaving Basic (list, rename, or delete). $179.95/$25.

ZEDIT

A character oriented text editor with 26 commands and "macro" capability for stringing multiple commands together. Included are a complete array of character move, add, delete, and display function. $49.95./$15.

ZTEL

Z80 Text Editing Language - Not just a text editor. Actually a language which allows you to edit text and also write, save, and recall programs which manipulate text. Commands include conditional branching, subroutine calls, iteration, block move, expression evaluation, and much more. Contains 36 value registers and 10 text registers. Be creative! Manipulate text with commands you write using Ztel. $79.95/$25.

TOP

A Z80 Text Output Processor which will do text formatting for manuals, documents, and other word processing jobs. Works with any text editor. Does justification, page numbering and headings, spacing, centering, and much more! $79.95/$25.

MACRO I

A macro assembler which will generate relocateable or absolute code for the 8080 or Z80 using standard Intel mnemonics plus TDL/Z80 extensions. Functions include 14 conditionals, 16 listing controls, 54 pseudo-ops, 11 arithmetic/logical operations, local and global symbols, chaining files, linking capability with optional linker, and recursive/reiterative macros. This assembler is so powerful you'll think it is doing all the work for you. It actually makes assembly language programming much less of an effort and more creative. $79.95/$20.

MACRO II

Expands upon Macro I's linking capability (which is useful but somewhat limited) thereby being able to take full advantage of the optional Linker. Also a time and date function has been added and the listing capability improved. $99.95/$25.

LINKER

How many times have you written the same subroutine in each new program? Top notch professional programmers compile a library of these subroutines and use a Linker to tie them together at assembly time. Development time is thus drastically reduced and becomes comparable to writing in a high level language but with all the speed of assembly language. So, get the new CDL Linker and start writing programs in a fraction of the time it took before. Linker is compatible with Macro I & II as well as TDL/Xitan assemblers version 2.0 or later. $79.95/$20.

DEBUG I

Many programmers give up on writing in assembly language even though they know their programs would be faster and more powerful. To them assembly language seems difficult to understand and follow, as well as being a nightmare to debug. Well, not with proper tools like Debug I. With Debug I you can easily follow the flow of any Z80 or 8080 program. Trace the program one step at a time or 10 steps or whatever you like. At each step you will be able to see the instruction executed and what it did. If desired, modifications can then be made before continuing. It's all under your control. You can even skip displaying a subroutine call and up to seven breakpoints can be set during execution. Use of Debug I can pay for itself many times over by saving you valuable debugging time. $79.95/$20.

DEBUG II

This is an expanded debugger which has all of the features of Debug I plus many more. You can "trap" (i.e. trace a program until a set of register, flag, and/or memory conditions occur). Also, instructions may be entered and executed immediately. This makes it easy to learn new instructions by examining registers/memory before and after. And a RADIX function allows changing between ASCII, binary, decimal, hex, octal, signed decimal, or split octal. All these features and more add up to give you a very powerful development tool. Both Debug I and II must run on a Z80 but will debug both Z80 and 8080 code. $99.95/$20.

ZAPPLE

A Z80 executive and debug monitor. Capable of search, ASCII put and display, read and write to I/O ports, hex math, breakpoint, execute, move, fill, display, read and write in Intel or binary format tape, and more! on disk

APPLE

8080 version of Zapple

NEW! TPM now available for TRS-80 Model II!

TPM*

A NEW Z80 disk operation system! This is not CP/M*. It's better! You can still run any program which runs with CP/M* but unlike CP/M* this operating system was written specifically for the Z80* and takes full advantage of its extra powerful instruction set. In other words its not warmed over 8080 code! Available for TRS-80* (Model I or II). Tarbell, Xitan DDDC, SD Sales "VERSA-FLOPPY", North Star (SD&DD), and Digital (Micro) Systems. $79.95/$25.

SYSTEM MONITOR BOARD (SMB II)

A complete I/O board for S-100 systems. 2 serial ports, 2 parallel ports, 1200/2400 baud cassette tape interface, sockets for 2K of RAM, 3-2708/2716 EPROM's or ROM, jump on reset circuitry. Bare board $49.95/$20.

ROM FOR SMB II

2KX8 masked ROM of Zapple monitor. Includes source listing $34.95/$15.

PAYROLL (source code only)

The Osborne package. Requires C Basic 2.
5" disks $124.95 (manual not included)
8" disks $ 99.95 (manual not included)
Manual $20.00

ACCOUNTS PAYABLE/RECEIVABLE (source code only)

By Osborne, Requires C Basic 2
5" disks $124.95 (manual not included)
8" $99.95 (manual not included)
Manual $20.00

GENERAL LEDGER (source code only)

By Osborne. Requires C Basic 2
5" disks $99.95 (manual not included)
8" disks $99.95 (manual not included)
Manual $20.00

C BASIC 2

Required for Osborne software. $99.95/$20.

SYSTEM/6

TPM with utilities, Basic I interpreter, Basic E compiler, Macro I assembler, Debug I debugger, and ZEDIT text editor.
Above purchased separately costs $339.75
Special introductory offer: Only $179.75 with coupon!!

$160.

This Coupon Is Worth One Hundred And Sixty Dollars Toward The Full Price Of The SYSTEM/6 Package
System/6 with this coupon is only $179.95.
This is a limited time offer.

$160.00

ORDERING INFORMATION

Visa, Master Charge and C.O.D. O.K. To order call or write with the following information.

1. Name of Product (e.g. Macro I)
2. Media (e.g. 8" CP/M)
3. Price and method of payment (e.g. C.O.D.) include credit card info. if applicable.
4. Name, Address and Phone number.
5. For TPM orders only: Indicate if for TRS 80, Tarbell, Xitan DDDC, SD Sales (5¼" or 8"). *ICOM* (5¼" or 8"), North Star (single or double density) or Digital (Micro) Systems.
6. N.J. residents add 5% sales tax.

Manual cost applicable against price of subsequent software purchase in any item except for the Osborne software.

For information and tech queries call
609-599-2146

For phone orders ONLY call toll free
1-800-327-9191
Ext. 676
(Except Florida)

OEMS

Many CDL products are available for licensing to OEMs. Write to Carl Galletti with your requirements.

* Z80 is a trademark of Zilog
* TRS-80 is a trademark for Radio Shack
* TPM is a trademark of Computer Design Labs. It is not CP/M*
* CP/M is a trademark of Digital Research

Prices and specifications subject to change without notice.

DEALER INQUIRIES INVITED.

COMPUTER DESIGN LABS

18

342 Columbus Avenue
Trenton, N.J. 08629

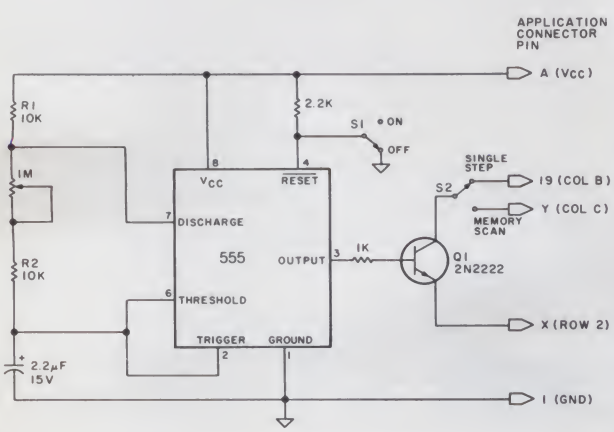
KIM Vari-Stepper

Set your own pace with an automatic single-stepper.

Larry M. Tannenbaum
PO Box 5013
Fullerton CA 92635

I have been using with my KIM-1 a simple and convenient circuit that complements the add-on described by Dr. Marvin L. De Jong in the September 1979 *Kilobaud Microcomputing* ("Catching Bugs with Lights," p. 96). His device automatically displays the contents of a selected register while in the single-step mode, without having to use the KIM keyboard. My contribution is a variable speed, automatic single-stepper. With both of these units installed, the KIM will step through a program at a rate you select, and you can watch a register value change without even touching that temperamental, built-in keypad.

The simple circuit is shown in the figure. I used a 555 timer to set the time interval between steps. The 1 meg potentiometer provides a good range of step rates: approximately 45 ms to 3 seconds. Three seconds is useful if you want to carefully check the program as it progresses and stop it at a specific address. The faster rates are useful for moving quickly through part of a program, such as a loop, when you don't need to see the actual progression.



Single-stepper circuit.

The step rate can be changed at any time by turning the potentiometer. If you would like a different range, experiment! A 2 meg potentiometer will give you about 6 seconds maximum between steps. If you want faster single-step execution, try smaller timing resistors (R1,R2) or a smaller capacitor.

There is a limit to how fast you can go. You must allow enough time between steps for the KIM to perform its single-step software and fully scan the keyboard; otherwise, erratic operation may result. I don't have a scope to check this out, and I didn't feel like tracing through the software to determine the amount of time KIM needs, so I can't tell you what it is.

Q1 can be any NPN switching transistor. I used a 2N2222. It is connected to the output of the timer and acts like a switch. When the 555's output goes high, the transistor shorts the "GO" key, fooling KIM into thinking that you pressed it. When the timer's output is low, the transistor is an open circuit, and KIM thinks you have released the key.

In order to use the stepper, you must set up the computer exactly as if you were manually single-stepping: The single-step switch must be on, and 1C00 must be stored in locations 17FB, 17FA. The auto-stepper is turned on and off by S1. When off, the "GO" key works as usual.

There is an extra bonus if you install switch S2. Connecting the collector of Q1 to application connector pin Y instead of 19 causes "+" key closures to be simulated. Now you can check a portion of memory, using a slow step rate, without pushing the "+" key a hundred or more times. This saves wear on my fingers and a lot of aggravation from bouncy keys.

The circuit can be built on any board that's convenient. Parts placement is not at all critical. The usefulness of this little gadget shows that good things do come in small packages! ■

The Personal Computer Line by OHIO SCIENTIFIC

Personal Computers

C1P: $349 A dramatic breakthrough in price and performance. Features OSI's ultra-fast BASIC-in-ROM, full graphics display capability, and large library of software on cassette and disk, including entertainment programs, personal finance, small business, and home applications. It's a complete programmable computer system ready to go. Just plug-in a video monitor or TV through an RF converter, and be up and running. 15K total memory including 8K BASIC and 4K RAM—expandable to 8K.

C1P MF: $995 First floppy disk based computer for under $1000! Same great features as the C1P plus more memory and instant program and data retrieval. Can be expanded to 32K static RAM and a second mini-floppy. It also supports a printer, modem, real time clock, and AC remote interface, as well as OS-65D V3.0 development disk operating system.

Professional Portables

C4P: $698 The professional portable that has over three times the display capability of C1Ps. Features 32 x 64 character display in up to 16 *colors*, graphics, audio output, a DAC for voice and music generation, key pad and joystick interfaces, AC remote control interface and much more. Utilizes a 4-slot BUS (2 used in base machine), 8K BASIC-in-ROM, 8K of *static* RAM and audio cassette interface. Can be directly expanded to 32K static RAM and two mini-floppy disks.

C4P MF: $1695 The ultimate portable computer has all the features of the C4P plus real time clock, home security system interface, modem interface, printer interface, 16 parallel lines and an accessory BUS. The standard machine operates at twice the speed of currently available personal computers (with GT option it runs even faster!). The C4P MF starts with 24K RAM and a single mini-floppy and can be directly expanded to 48K and two mini-floppies. Available software includes games, personal, business, educational and home control applications programs as well as a real time operating system, word processor and a data base management system.

Computers come with keyboards and floppies where specified. Other equipment shown is optional.

Home/Small Business Systems

C8P: $895 Same great features as the C4P in a tremendously expandable "mainframe package." Features over three times the expansion capability of the C4P for advanced home and demanding business applications. Can be expanded to 48K RAM, dual 8″ floppies, hard (Winchester) disks and multiple I/O devices such as Voice I/O and a universal telephone interface.

C8P DF: From $2597 The ultimate Home/Very Small Business Computer at a personal computer price. Features 32K RAM (expandable to 48K) and dual 8″ floppy disks (stores eight times as much information as a mini-floppy). Has all personal computer capabilities including 32 x 64 display, color graphics, sound, DAC, joystick interfaces, home features including real time clock, AC remote interface, home security and fire detection interface and can be expanded to include voice I/O and a universal telephone system for answering and initiating calls! Its large memory capability and 8″ floppies allow it to run most Ohio Scientific business system software including a compete accounting system, word processor and information management system.

For literature and the name of your local dealer, CALL 1-800-321-6850 TOLL FREE.

OHIO SCIENTIFIC
1333 SOUTH CHILLICOTHE ROAD
AURORA, OH 44202 • (216) 831-5600

Lee Wilkinson
Rte. 5, Wilkinson Pike
Maryville TN 37801

A Poor Man's Computer Paintbrush

Have you ever watched the man in the computer portrait booth transfer images of people's faces into two-dimensional computer portraits? You probably said, "Boy, I sure would like to do a picture like that on my computer." If you were as bold as I was, you probably inquired about the cost of this magical device. Finding that the $15,000 price tag was a little out of your hobby budget, especially since the portrait computer couldn't play Star Trek, you probably decided that the art world would never be availed of your electronic-artistic genius.

Well, don't despair. This article describes how you, too, can "draw" computer portraits even without a TV camera. You will need, however, some type of hard-copy device. Anything from a TTY terminal on up will be sufficient. The wider the column width of your printer, the larger you can make your portrait.

You may not be able to produce one of these original "works of art" quite as fast as your local computer portrait studio counterpart, but think of the fun you'll have when friends come over and amazingly you sit down to play . . . a computer portrait.

Getting Started

The first step is to select a portrait or a picture that you wish to reproduce with your computer (Photo 1). Make a couple of non-glossy photographic copies from the original (it helps to have a friend with a darkroom if you don't do the photographic work yourself) on the highest-contrast photographic paper that you (or your friend) have available.

Since normal photographic paper can produce as many as ten or eleven gray tones, the high-contrast paper "drops" out some of these tones (Photo 2). This allows you to more closely approximate the six tones you will be able to reproduce. An 8 × 10 workprint seems to be a convenient working enlargement for the steps to follow. The paper that I use is Kodak photomechanical transfer (PMT), which is a stabilization-type paper. However, any high-contrast photographic paper will reduce the tonal range just fine.

Making a Grid

Take one of your workprints and divide the 8 inch side (assuming you are using an 8 × 10) into 60 divisions. You can use drafting instrument dividers, compasses or whatever means are at your disposal. Using a sharp-pointed, soft lead pencil, draw parallel lines on your workprint defining the divisions you just made.

Now make 60 divisions on the 10 inch side. Draw parallel lines for these divisions also. This produces a 60 × 60 grid on your workprint. Referring to the tonal chart (Table 1), you can now start with each line and code each box on your workprint with one of the tonal characters.

Determining Tone Codes

You will notice in Table 1 that the six gray tones range from X (the darkest) to S (the lightest).

** W I N T E R W O N D E R L A N D **
BY LEE WILKINSON -- MARYVILLE, TN.

X—The darkest tone that can be produced (made by printing Ms and issuing a carriage return without a line feed, then overprinting the Ms with Ws).
M—The next lighter tone.
I—A mid-range tone (sometimes changed to the letter C with the ASCII conversion characters in the main program).
:—A lighter mid-range tone designated as the letter K in the data statements to differentiate from carriage returns.
.—Next to the lightest tone in the gray scale. Designated by the letter P in the data statements.
space—The lightest tone produced. Designated as the letter S in data statements. Used very sparingly for highlights, it gives dimension and depth to the portrait.
C—Carriage return used to overprint a line with Ws to produce the darkest tone before a line feed.
L—Line feed, used after all printing on a current line is completed.
E—End flag to signal completion of portrait data.

Table 1. Symbols used in data statements.

The symbols M, I, : and . produce the other four intermediate tones. This makes a total of six gray tones with which we can create an illusion on the two-dimensional paper.

Keeping in mind that we will be working with these six tones, fill in each of the 3600 boxes on your workprint grid (Photo 3) with appropriate tonal characters. At first it may seem like a monumental task, but you'll soon get the hang of it. These boxes will be used to construct data statements. As you can see by examining the program listings, these data statements constitute the bulk of our program.

Photo 1. The original portrait with the full tonal range.

Translating to Data Statements

Make a scale from a stiff piece of cardboard and graduate it with the widths of your boxes. These marks should be equal in size with the grid (Photo 4) of your workprint. Number these divisions consecutively starting at number 1. The last number should be equal to the widest carriage position that your picture will occupy.

Using the 60 × 60 grid, the scale will have 60 divisions. This scale will aid you greatly in counting tonal characters. Starting with the first row on your workprint, count the number of each tonal character. Work from left to right and record each occurrence of the tonal characters.

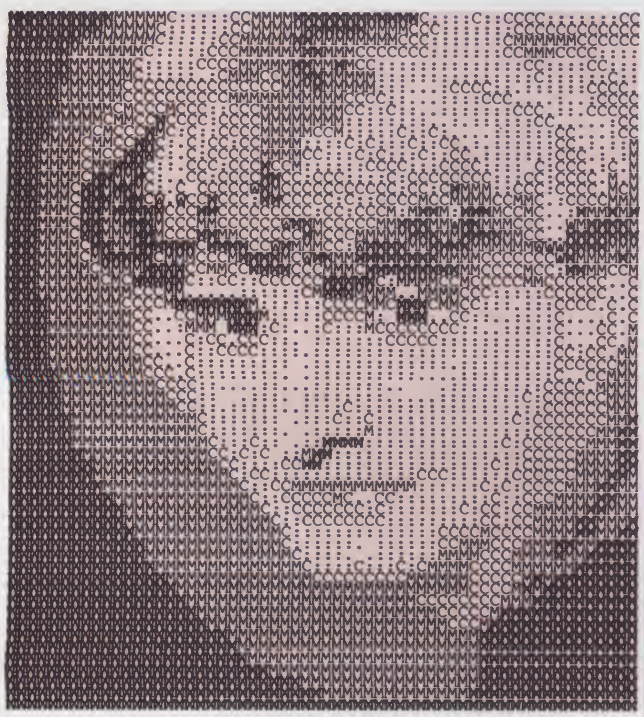
Note that in the first row (Photo 3) there are 15 Ms. (Remember that X is made by Typing W over M.) Next, there are 1-I, 5-K (colons), 2-I, 16-M, 1-I, 4-K, 1-I, 2-K, 4-I, 7-K, 2-I and 1-C. The 1-C allows a carriage return without a line feed so that we may overprint the Ms with Ws to make the darkest tone.

Now that the carriage is returned we print ten Ws (over the first ten of the 15 Ms), 17 S (spaces), 12 W and then 1 C and 1 L for the carriage return and the line feed to start the next line.

Refer to Listing 1 at the first data statement to see that this is exactly the way the first data statement is written. Continue writing your data statements in this manner until your entire portrait has been translated into data statements. Your last line should contain the flag 1E. This is necessary in order to signal your main program that all data has been read; otherwise, an error could occur.

The Main Program

Lines 20 and 25 of the program (Listing 1) simply com-



** ROSE ANN WILKINSON **
PROGRAMED ON AN ALTAIR 8800 COMPUTER
BY LEE WILKINSON -- MARYVILLE, TN.

Photo 2. Copy of the original. This photo workprint is made on high-contrast paper. This copy is used to reduce the number of tones in the portrait aiding in the selection of the six tonal range codes that are used to reproduce the final computer portrait.

Photo 4. Translation of the tonal codes into data statements. Using "scaler" to aid in counting the occurrences of the tonal codes with which the data statements are constructed.

mand your hard-copy device to do the printing. You may need a couple of nulls here also. Some hard-copy devices require these nulls in order to delay printing until the carriage is returned and realigned to the first print position.

Line 30 reads the portrait from the data statements. Lines 40 through 60 split the data string into the variables A and A$. A then becomes the number of times the tone code will be printed. The tone code has also been assigned to A$.

Lines 70 through 107 reassign A$ to the ASCII characters that you desire to print. For example, line 90 sets A$ to ASCII 46, which is a period (.). Line 107 sets A$ to an ASCII 67, which is the letter C, rather than the originally coded I. I find this gives a smoother intermediate tone for this particular portrait. By changing the ASCII codes in lines 70 through 107, you can extensively alter the results of the portrait. You can actually produce a negative portrait by reversing the ASCII codes.

Line 110 is the condition for the end of the data statements. The for/next loop at lines 120 through 140 do the repetitive printing, and line 150 repeats the whole sequence.

Line 160 restores the nulls to zero in the event the main terminal can operate without the delays. Line 210 is used to resume the I/O on the main terminal rather than the hard-copy device.

Additional Programs

Listing 2, "Winter Wonderland," took me approximately 30 hours to translate into data statements. It takes a 110 baud printer 20 to 30 minutes to print the picture. Winter Wonderland is especially beautiful when overlaid with a transparent light blue sheet of plastic, matted with a black mat, framed, and viewed from 5 feet.

My original intentions were to send copies of this computer art for Christmas cards to computer friends around the country. However, like many of my other projects, this one has been temporarily shelved for a few months.

Conclusion

Both of the programs are written in a Digital Equipment-type of BASIC. Simple conversions of the data split lines will allow the program to run on a North Star-type of BASIC.

This method of computer portraits is admittedly slow. The advantage of this method allows you control over the final picture to darken edges, lighten areas or even abstractions on the portrait. That you can have a personalized portrait that you created makes this program worth investigating at least a few times. I'll just bet that you place some of your computer portraits on your walls, too. ■

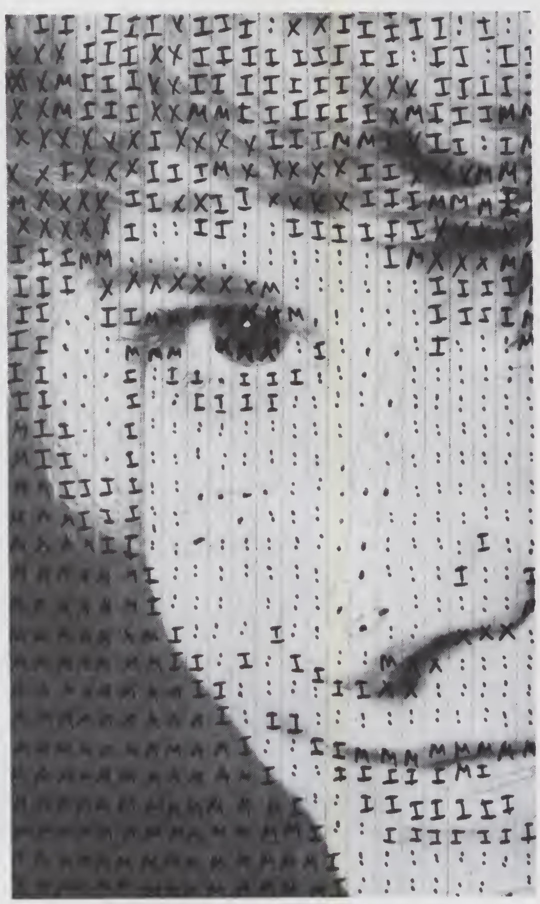


Photo 3. The photo workprint with 60 × 60 grid lines drawn and tonal codes written in the boxes.

Listing 1. Portrait program.

```
REM       ***    ROSE ANN WILKINSON    *** \
                   a computer portrait
             by Lee Wilkinson, (615) 982-6703    /

20    REM put nulls here if necessary to delay printer

25    LPRINTER:Rem switch printer on-line
30    READ A$

REM     SPLIT DATA INTO "A" AND "A$"

        REM  Is "A$" 3 characters long ?
40    IF LEN(A$)=3 THEN A=VAL(LEFT$(A$,2)):GOTO 60

        REM  No, just 2 characters long
50    A=VAL(LEFT$(A$,1))

        REM What character shall we use ?
60    A$=RIGHT$(A$,1)

REM     CHANGE "A$" TO ASCII IF NECESSARY

70    IF A$="C" THEN A$=CHR$(13):GOTO 120:REM <cr>
80    IF A$="L" THEN PRINT:GOTO 30:REM <lf>
90    IF A$="p" THEN A$=CHR$(46):GOTO 120:REM "PERIOD"
100   IF A$="S" THEN A$=CHR$(32):GOTO 120:REM "SPACE"
105   IF A$="K" THEN A$=CHR$(58):GOTO 120:REM "COLON"
```

```
107  IF A$="I" THEN A$=CHR$(67):GOTO 120:REM CHANGE "I" TO "C"
110  IF A$="E" THEN GOTO 160:REM END FLAG

REM  NOW PRINT 'A$', 'A' TIMES

120  FOR I%= 1 TO A
130  PRINT A$;
140  NEXT I%

150  GOTO 30:REM read the next statement

160 REM reset nulls for main terminal here, if necessary
170 PRINT:PRINT:PRINT TAB(12);"**   R O S E   A N N   W I L K I N S O N
180 PRINT:PRINT TAB(15);"PROGRAMED ON AN ALTAIR 8800 COMPUTER"
190 PRINT:PRINT TAB(16);"BY LEE WILKINSON -- MARYVILLE, TN."
      REM line feed up to tear off
200 FOR I = 1 TO 15:PRINT:NEXT I

210 CONSOLE:rem Put routine to return to main terminal

REM Data statements start here
            REM  Line #1
1000 DATA 15M,1I,5K,2I,16M,1I,4K,1I,2K,4I,7K,2I,1C,10W,17S,12W,1C,1L
            REM  Line #2
1010 DATA 14M,1I,6K,2I,15M,3I,7K,12I,1C,9W,19S,6W,1C,1L
            REM  Line #3
1020 DATA 14M,1I,5K,2I,15M,3I,7K,1I,6M,6I,1C,7W,20S,2W,1C,1L
1030 DATA 13M,1I,5K,3I,11M,7I,7K,2I,3M,4I,4K,1C,6W,22S,2W,1C,1L
1040 DATA 12M,1I,5K,4I,11M,2I,14K,2I,4K,2I,3K,1C,6W,21S,2W,2S,1W,1C,1L
1050 DATA 12M,1I,5K,1P,1K,1I,3M,2I,9M,15K,1I,6K,1I,2K,1C,6W,22S,2W
1051 DATA 1C,1L
1060 DATA 11M,15I,9M,3K,3P,1K,4I,2K,1P,6K,4I,1P,1C,5W,24S,1W,1C,1L
1070 DATA 11M,10I,11M,4I,2K,3P,4K,3I,4P,1K,2P,1K,4I,1C,5W,1C,1L
1080 DATA 10M,1I,1M,3I,1M,3I,2K,3I,8M,2I,2K,1I,2K,2P,7K,3I,1P,2K,1P,1I
1081 DATA 4I,1C,5W,1C,1L
1090 DATA 9M,1I,2M,2I,1M,4I,1K,4I,8M,8K,2P,1K,1P,6K,2I,5P,1K,2I
1091 DATA 1C,4W,10S,1W,1C,1L
1100 DATA 8M,1I,1M,3I,2M,1I,1K,1I,2K,4I,5M,2I,1M,5K,1I,2K,1I,1K,4P,6K
1101 DATA 2I,3P,1K,2I,1C,4W,9S,1W,1C,1L
1110 DATA 7M,1I,2M,2I,2M,1I,1K,1P,1I,1K,2I,1K,2I,4M,4K,1I,1K,1I,3K
1111 DATA 1I,2K,2I,1P,1K,1P,6K,2I,1P,1K,1P,2K,1M,1C,3W,9S,2W,1C,1L
1120 DATA 7M,1I,1M,2I,3M,1I,1K,1P,1I,1K,5I,4M,2I,3K,1I,1K,1I,3K,1I
1121 DATA 2K,2I,1P,7K,3I,1P,1K,1P,1K,1M,1C,3W,8S,3W,1C,1L
1130 DATA 7M,1I,5M,1I,2K,1P,3I,1K,3I,1M,2I,1M,6K,4I,3K,4I,4K,1S,1I
1131 DATA 1K,3I,1P,1K,1I,1P,1M,1C,3W,7S,3W,11S,1W,1C,1L
1140 DATA 7M,1I,5M,1I,1K,2P,2I,1K,4I,2M,2I,1K,3I,4K,1I,1K,8I,3K,2I
1141 DATA 1K,4I,1P,1M,1P,1M,1C,3W,5S,2W,2S,1W,11S,2W,1C,1L
1150 DATA 6M,1I,6M,1I,1K,1P,1I,1S,6I,2M,11I,1K,2I,1K,1I,4M,2K,3I
1151 DATA 1K,4I,1P,1M,1I,1M,1C,3W,4S,2W,1S,2W,11S,2W,17S,1W,1C,1L
1160 DATA 6M,1I,6M,2I,1P,3I,1M,3I,1K,2M,5I,1K,1I,2K,4I,1M,2I,4M
1161 DATA 2I,2M,1I,1P,1K,4I,3M,1C,3W,4S,3W,1S,2W,6S,1W,4S,2W,17C
1162 DATA 3W,11S,1W,1S,1W,1C,1L
1170 DATA 6M,1I,8M,3I,2M,9I,1K,2I,1K,3I,1M,1K,4M,1K,4M,2I,1M,1I
1171 DATA 2P,1K,6M,1C,3W,4S,2W,1S,5W,3S,2W,19S,2W,2S,3W,8S,1W,2S
1172 DATA 1W,1S,1W,1C,1L
1180 DATA 6M,1I,8M,3I,2M,7I,3M,5I,7M,1K,5M,3I,2P,1K,6M,1C,3W,4S
1181 DATA 2W,2S,3W,12S,3W,8S,3W,2S,3W,8S,1W,1S,4W,1C,1L
1190 DATA 6M,1I,8M,3I,4M,6I,2M,3I,16M,2I,2P,7M,1C,3W,4S,2W,3S,2W
1191 DATA 4S,2W,8S,1W,9S,3W,2S,4W,6S,7W,1C,1L
1200 DATA 7M,1I,10M,1I,4M,3I,4M,2I,1K,1I,16M,1I,2K,7M,1C,3W,5S,2W
1201 DATA 2S,6W,1S,4W,2S,2W,6S,5W,2S,4W,6S,7W,1C,1L
1210 DATA 7M,1I,6M,1I,3M,3I,6M,2I,16M,7I,1K,7M,1C,3W,6S,5W,8S,4W,2S
1211 DATA 3W,2S,8W,11S,2W,4S,1W,1C,1L
1220 DATA 8M,1I,8M,2I,2M,2I,4M,3I,3M,1I,11M,4I,1M,2I,1K,4M,2I,1M,1C
1221 DATA 3W,10S,4W,6S,4W,7S,9W,10S,2W,1C,1L
1230 DATA 9M,1I,1M,1I,5M,1I,2K,2I,2K,6I,5M,1I,1K,4M,2I,1M,5I,2M,6I
1231 DATA 2P,1I,1C,3W,9S,5W,13S,5W,1C,1L
1240 DATA 12M,3I,2M,5P,6K,1I,8M,2P,4M,3I,5K,6I,2P,1I,1C,3W,27S
1241 DATA 3W,1C,1L
1250 DATA 11M,4I,9M,6K,4I,2M,1I,3M,1I,2M,2I,3K,1P,3K,4I,1P,1I,1P
1251 DATA 1I,1C,4W,12S,7W,14S,3W,1C,1L
1260 DATA 12M,2I,2P,2I,7M,5K,4I,1M,1K,1S,3M,2P,2I,2K,2P,4K,1I,4P
1261 DATA 1I,1P,1I,1C,4W,17S,3W,13S,3W,1C,1L
1270 DATA 12M,2I,2P,1K,3M,1S,3M,1K,1I,1K,2P,1K,1I,3K,1M,8I,2K,2P
1271 DATA 5K,1I,3P,2I,1P,1M,1C,4W,17S,3W,1C,1L
1280 DATA 12M,2I,3P,1I,1K,2I,1P,3I,2K,1P,1S,7K,5I,11K,1I,3P,2I,1P
1281 DATA 1M,1C,4W,1C,1L
1290 DATA 12M,2I,3P,1I,2K,4I,4K,1P,23K,2I,1K,2I,1P,2M,1C,4W,1C,1L
1300 DATA 13M,2I,2P,1I,9K,1S,1P,13K,1P,9K,6I,2M,1C,4W,1C,1L
1310 DATA 13M,3I,1P,1I,8K,1P,1S,1P,10K,4P,9K,1I,2P,2I,3M,1C,5W,1C,1L
1320 DATA 14M,4I,2K,4P,2K,3P,7K,2P,3S,1P,9K,4I,1P,4M,1C,5W,1C,1L
1330 DATA 15M,3I,2K,2S,1P,3K,1P,1S,8K,5P,8K,1I,1K,6I,3M,1C,5W,1C,1L
1340 DATA 16M,2I,2K,1P,5K,2P,4K,1I,18K,6I,3M,1C,5W,1C,1L
1350 DATA 18M,1I,7K,1S,1P,3K,1I,2K,1I,13K,1I,1K,7I,3M,1C,6W,53S,1W,1C
1351 DATA 1L
1360 DATA 18M,1I,7K,1P,1S,1P,5K,1M,15K,6I,4M,1C,6W,52S,2W,1C,1L
1370 DATA 19M,1I,3K,1I,2K,1S,1P,2K,4M,13K,1I,1K,6I,5M,1C,6W,24S,4W
1371 DATA 24S,2W,1C,1L
1380 DATA 19M,2I,1K,1I,1K,1I,1S,2K,3M,18K,5I,6M,1C,7W,22S,2W,27S,2W,1C
1381 DATA 1L
1390 DATA 20M,2I,4K,2I,2M,16K,1I,2K,5I,6M,1C,8W,20S,2W,28S,2W,1C,1L
1400 DATA 21M,1I,1K,2I,14K,3I,6K,6I,6M,1C,8W,49S,2W,1C,1L
1410 DATA 22M,1I,2K,2I,12M,6K,1I,1K,1I,1K,4I,7M,1C,9W,47S,2W,1C,1L
1420 DATA 23M,1I,2K,5I,1M,1I,2P,2I,9K,1I,2K,3I,8M,1C,10W,46S,2W,1C,1L
1430 DATA 24M,1I,2K,7I,1K,1I,7K,1I,1K,2I,1K,2I,10M,1C,11W
1431 DATA 44S,2W,1C,1L
1440 DATA 25M,1I,2K,8I,9K,2I,1K,2I,10M,1C,11W,43S,2W,3S,1W,1C,1L
1450 DATA 26M,1I,14K,4I,1M,4I,10M,1C,12W,42S,2W,2S,2W,1C,1L
1460 DATA 26M,1I,6K,1P,1S,6K,2I,3M,3I,11M,1C,13W,39S,2W,2S,4W,1C,1L
1470 DATA 27M,1I,4K,1P,1S,1P,6K,4M,3I,12M,1C,13W,39S,1W,2S,5W,1C,1L
1480 DATA 28M,1I,4K,1I,3K,1I,1K,1I,4M,4I,12M,1C,14W,40S,6W,1C,1L
1490 DATA 29M,9I,6M,5I,11M,1C,15W,35S,1W,1S,8W,1C,1L
1500 DATA 44M,4I,12M,1C,16W,33S,1W,1S,9W,1C,1L
1510 DATA 39M,8I,13M,1C,17W,33S,10W,1C,1L
1520 DATA 41M,5I,14M,1C,18W,29S,13W,1C,1L
1530 DATA 42M,3I,15M,1C,19W,27S,14W,1C,1L
1540 DATA 60M,1C,21W,24S,15W,1C,1L
1550 DATA 60M,1C,22W,23S,15W,1C,1L
1560 DATA 60M,1C,23W,22S,15W,1C,1L
1570 DATA 60M,1C,25W,20S,15W,1C,1L
1580 DATA 60M,1C,28W,17S,15W,1C,1L
1590 DATA 60M,1C,30W,14S,16W,1C,1L
1600 DATA 60M,1C,60W,1C,1L
10000 DATA 1E
END
```

Listing 2. Winter Wonderland.

```
REM    ***   WINTER WONDERLAND   ***  by Lee Wilkinson

REM    put nulls here if needed

25     LPRINTER:REM switch printer on-line
30     READ A$
REM    Split data into A and A$
40     IF LEN(A$)=3 THEN A=VAL(LEFT$(A$,2)):GOTO 60
50     A=VAL(LEFT$(A$,1))
60     A$=RIGHT$(A$,1)
70     IF A$="C" THEN A$=CHR$(13):GOTO 120:REM <cr>
80     IF A$="L" THEN PRINT:GOTO 30:       REM <lf>
90     IF A$="P" THEN A$=CHR$(46):GOTO 120:REM "PERIOD"
100    IF A$="S" THEN A$=CHR$(32):GOTO 120:REM "SPACE"
110    IF A$="E" THEN 160
120           FOR I%=1 TO A
              PRINT A$;
140           NEXT I%
150    GOTO 30

160     REM    reset nulls for CRT here

170    PRINT:PRINT:PRINT TAB(15);"**  W I N T E R   W O N D E R L A N D  **"
190    PRINT:PRINT TAB(19);"BY LEE WILKINSON -- MARYVILLE, TN."
200           FOR I=1 TO 15
              PRINT
              NEXT I

210    CONSOLE : REM BACK TO CRT

REM    data statements start here

1000 DATA 14B,56I,1C,1L,11B,59I,1C,1L,11B,59I,1C,1L,11B,57I,2B,1C,1L,8B
1001 DATA 60I,2B,1C,1L
1010 DATA 9B,60I,1B,1C,1L,9B,58I,3B,1C,68S,2W,1C,1L
1020 DATA 9B,59I,2B,1C,1L,10B,58I,2B,1C,2W,66S,2W,1C,1L
1030 DATA 10B,55I,5B,1C,1W,65S,4W,1C,1L,10B,13I,5P,39I,3B,1C,1W,66S,3W
1031 DATA 1C,1L
1040 DATA 8B,13I,6P,40I,3B,1C,1W,66S,3W,1C,1L,5B,31I,2P,29I,3B,1C,2W
1041 DATA 1S,2W,62S,1W,1S,1W,1C,1L
1050 DATA 4B,33I,4P,27I,2B,1C,4W,64S,2W,1C,1L,4B,1I,1B,32I,5P
1051 DATA 24I,1B,1C,1S,2W,64S,1W,1C,1L
1060 DATA 3B,1I,3B,30I,9P,21I,1S,2B,1C,1S,1W,4S,1W,61S,2W,1C,1L
1070 DATA 6B,32I,10P,7I,1B,7I,1S,2I,4B,1C,1S,2W,2S,1W,60S,2W,1S,1W,1C
1071 DATA 1L
1080 DATA 6B,35I,11P,3I,1B,1I,1B,7I,5B,1C,2W,2S,2W,51S,1W,11S,1W
1081 DATA 1C,1L
1090 DATA 5B,1I,1B,3I,3B,31I,9P,4I,3B,8I,2B,1C,3W,1S,1W,1S,1W
1091 DATA 52S,1W,1C,1L
1100 DATA 3B,2I,2B,6I,3B,29I,4P,7I,3B,9I,2B,1C,3W,2S,2W,50S,1W,10S
1101 DATA 1W,1C,1L
```

```
1110 DATA 3B,1I,2B,8I,5B,37I,3B,7I,1B,1I,2B,1C,3W,1S,2W,51S,2W,7S,1W
1111 DATA 1S,1W,1C,1L
1120 DATA 6B,7I,7B,36I,2B,1I,1B,5I,5B,1C,2S,4W,50S,2W,7S,2W,1C,1L
1130 DATA 5B,4I,11B,36I,3B,5I,6B,1C,1S,4W,51S,1W,8S,2W,1C,1L
1140 DATA 19B,36I,3B,7I,5B,1C,2W,53S,1W,11S,1W,1C,1L
1150 DATA 20B,31I,8B,9I,2B,1C,1W,2S,2W,49S,5W,9S,2W,1C,1L
1160 DATA 2B,2I,12B,37I,2P,15B,1C,2W,2S,2W,51S,2W,5S,1W,3S,1W,1C,1L
1170 DATA 17B,35I,1P,17B,1C,5W,1S,1W,48S,2W,1S,4W,2S,1W,2S,3W,1C,1L
1180 DATA 17B,38I,11B,1P,3B,1C,1S,4W,1S,1W,48S,1W,1S,1I,5S,3W,1S,1W
1181 DATA 1C,1L
1190 DATA 6B,1P,1B,1P,7B,4I,2B,31I,16B,1C,1S,3W,1S,1W,52S,1W,1S,2W
1191 DATA 4S,2W,1C,1L
1200 DATA 5B,2P,1B,5P,6B,2I,4B,26I,19B,1C,3W,1S,1W,50S,2W,4S,1W,2S
1201 DATA 2W,1C,1L
1210 DATA 5B,1P,2B,6P,14B,2I,2B,22I,2B,1I,9B,3P,1B,1C,2W,1S,2W,49S
1211 DATA 2W,1S,1W,5S,3W,1C,1L
1220 DATA 4B,2P,1B,9P,17B,14I,3B,1P,3I,6B,2I,3B,1C,1S,3W,48S,3W
1221 DATA 1S,3W,3S,3W,1C,1L
1230 DATA 4B,1P,2B,12P,14B,17I,4B,2P,1B,2I,11B,1C,2S,2W,46S,4W,2S
1231 DATA 1W,2S,7W,2S,2W,1C,1L
1240 DATA 6B,16P,12B,19I,1S,1P,2B,3I,2B,5I,2P,1C,1W,1S,2W,1S,1W
1241 DATA 50S,1W,1C,1L
1250 DATA 7B,4P,1B,12P,15B,12I,2S,1P,4B,3I,5B,4P,1C,2W,2S,2W,40S
1251 DATA 1W,7S,1W,1S,1W,8S,1C,1L
1260 DATA 8B,2P,2B,14P,13B,11I,2S,1P,3B,1P,1S,2B,1I,4B,3P,2I,1C
1261 DATA 2W,1S,2W,42S,1W,5S,1W,1S,1W,5S,1W,2S,1W,1C,1L
1270 DATA 4B,2I,6B,16P,13B,5I,1B,6P,4B,3P,4B,2P,4I,1C,4W,4S
1271 DATA 1W,2S,1W,34S,1W,8S,1W,6S,2W,2S,1W,1C,1L
1280 DATA 4B,1I,6B,4I,18P,9B,4I,9B,1P,2B,2P,10B,1C,1S,3W,3S,1W
1281 DATA 2S,1W,35S,1W,1S,1W,5S,1W,5S,6W,1C,1L
1290 DATA 3B,2I,6B,7I,21P,7B,1I,7B,2P,4B,1P,4B,1S,4B,1C,3W,3S,2W
1291 DATA 1S,2W,37S,2W,2S,1W,4S,2W,1S,1W,3S,2W,1C,1L
1300 DATA 10B,9I,23P,11B,2P,3B,1P,10B,1P,1C,2W,4S,4W,38S,5W,3S
1301 DATA 2W,4S,2W,3S,2W,1C,1L
1310 DATA 7B,13I,4P,2I,21P,4B,2P,5B,2P,7B,1S,2P,1C,7W,41S,3W
1311 DATA 5S,2W,5S,1W,2S,1W,1C,1L
1320 DATA 2B,1I,1B,8P,10I,3P,3I,18P,5B,1S,3B,1P,3B,2P,5B,2S
1321 DATA 2P,1C,2W,1S,1W,44S,1W,1S,1W,8S,1W,2S,1W,1C,1L
1330 DATA 6B,2P,2B,3P,3I,9P,4I,17P,7B,2P,1S,4B,2P,4B,2S,1P,1I,1C
1331 DATA 6W,2S,2W,37S,2W,1S,1W,7S,2W,5S,1W,1C,1L
1340 DATA 3B,1P,5B,5P,3I,9P,5I,13P,1B,1P,7B,2P,1S,10B,2S,2I,1C
1341 DATA 1S,2W,1S,5W,38S,1W,1S,3W,4S,2W,1S,6W,1C,1L
1350 DATA 4B,2I,1B,1I,6P,6I,8P,5I,11P,7B,1P,2S,2I,2B,2P,4B,1P,1B
1351 DATA 2S,2I,1C,2S,2W,2S,1W,40S,1W,1S,1W,6S,2W,7S,1W,1C,1L
1360 DATA 10B,6P,6I,8P,5I,10P,8B,2I,3B,2P,4B,2S,2P,2B,1C,3S,7W,36S
1361 DATA 2W,7S,1W,1S,1W,1C,1L
1370 DATA 13B,1I,4P,6I,7P,6I,3P,5B,1P,2B,1I,3B,3P,4B,1S,1I,4B,1P,2I
1371 DATA 1C,13W,28S,1W,7S,1W,8S,1W,1S,4W,1C,1L
1380 DATA 6B,2I,5B,8P,5I,10P,6I,2P,1B,1P,4B,3S,11B,2S,2B,1P,1B,1C,3W,2S
1381 DATA 1W,2S,1W,1S,2W,32S,1W,1S,4W,3S,4W,2S,2W,1S,2W,5S,1W,1C,1L
1390 DATA 6B,2P,2B,2I,26P,4I,5B,3S,1P,2B,1P,1I,6B,1S,1P,2B,3P,1C,6W
1391 DATA 3S,2W,33S,2W,1S,2W,4S,2W,2S,1W,2S,1W,1S,1W,1C,1L
1400 DATA 11B,5I,1P,1I,2P,1I,21P,3I,3B,1P,1S,2P,4B,1P,2I,3B,2P,1S,2B
1401 DATA 2P,1I,1C,1W,2S,1W,2S,4W,35S,3W,5S,1W,1S,1W,5S,1W,1C,1L
1410 DATA 11B,11I,2P,1I,4P,1I,5P,1I,2P,1I,5P,3B,3P,6B,2P,3I,5B,3P,1I
1411 DATA 1C,1S,1W,1S,2W,40S,2W,5S,2W,1S,1W,1C,1L
1420 DATA 9B,13I,2P,3I,1P,8I,1P,1I,2P,13B,4P,3I,4B,2P,3I,1C,1S,1W,1S
1421 DATA 4W,35S,3W,2S,2W,3S,1W,10S,1W,4S,1W,1C,1L
1430 DATA 3B,41I,9B,1S,1I,1B,6P,1I,3B,2P,2I,1C,1S,1W,4S,1W,37S,1W,1S
1431 DATA 1W,2S,1W,1S,1W,11S,2W,1C,1L
1440 DATA 4B,33I,2P,4I,9B,2S,1I,2B,3P,2B,2P,1B,2P,2I,1B,1C,2W,41S,1W
1441 DATA 3S,4W,8S,1W,1C,1L
1450 DATA 5B,1I,1B,2I,3B,22I,3S,3P,2I,10B,2P,4B,2P,4B,1P,1I,4B
1451 DATA 1C,1W,1S,1W,3S,1W,2S,3W,30S,7W,8S,1W,1C,1L
1460 DATA 1I,2B,1S,7B,14I,1S,1P,3I,6S,5P,1I,9B,1S,1I,3P,7B,1P,1I,4B
1461 DATA 1C,2S,1W,3S,5W,31S,3W,2S,2W,9S,1W,1S,3W,5S,1W,1C,1L
1470 DATA 1I,8B,2P,13I,2S,1P,9S,1P,2B,2P,1I,8B,1S,1P,2S,2P,1B,1P,1B
1471 DATA 3P,3B,3P,1S,1B,1C,1S,2W,3S,3W,29S,1W,3S,5W,9S,1W,1S,1W,3S
1472 DATA 3W,4S,1W,1C,1L
1480 DATA 10B,3P,11I,1S,2P,9S,4B,2P,6B,5P,2S,3B,3P,8B,1C,2S,4W,1S,1W
1481 DATA 29S,3W,3S,5W,7S,3W,5S,6W,1C,1L
1490 DATA 4B,4I,7B,8I,1S,1P,1B,2P,6S,1P,5B,2P,4B,4P,2S,5B,4P,5B,2S
1491 DATA 2P,1C,1S,3W,6S,2W,13S,1W,10S,1W,2S,1W,5S,1W,8S,3W,6S,3W,1C,1L
1500 DATA 5B,4I,7B,7I,1P,3B,2P,4S,2P,6B,2P,13B,1S,4P,3B,1P,2S,2P,1B,1C
1501 DATA 1W,1S,3W,6S,3W,10S,1W,1S,1W,13S,1W,12S,3W,7S,1W,1C,1L
1510 DATA 8B,2I,2B,2I,1B,7I,7B,4S,1P,7B,3P,10B,3S,3P,1B,1P,1B,2P
1511 DATA 5B,1C,1W,3S,4W,6S,1W,12S,2W,15S,2W,3S,1W,2S,2W,6S,1W,1S,1W,2S
1512 DATA 1W,2S,2W,1C,1L
1520 DATA 9B,14I,1S,1B,4S,1P,10B,2P,8B,5S,1P,11B,1C,1W,2S,6W,32S,2W
1521 DATA 2S,2W,6S,8W,1S,2W,1C,1L
1530 DATA 11B,11I,2S,4B,3S,1P,20B,3S,1P,2I,9B,1S,1B,1P,1C,2W,2S,2W
1531 DATA 1S,4W,24S,3W,1S,3W,2S,8W,6S,1W,1S,1W,2S,1W,2S,1W,1C,1L
1540 DATA 11B,1P,9I,10S,1P,17B,3S,4P,1B,1I,8B,2S,2P,1C,2W,3S,1W,4S
1541 DATA 1W,24S,3W,1S,3W,2S,5W,12S,2W,1C,1L
1550 DATA 12B,5P,3I,5S,2B,1S,2B,1S,13B,2P,1B,7P,2I,9B,2S,2P,1B,1C
1551 DATA 3W,2S,3W,17S,2W,1S,2W,1S,1W,3S,3W,1S,3W,4S,1W,13S,2W,1C,1L
```

```
1560 DATA 16B,5I,1S,21B,3P,3S,1I,2P,1B,2I,7B,3S,1P,1S,1P,2B,1C
1561 DATA 1S,3W,2S,3W,13S,1W,2S,2W,1S,2W,1S,1W,26S,3W,1C,1L
1570 DATA 13B,9I,24B,1P,1S,3I,9B,4S,4P,2B,1C,1S,10W,11S,1W,2S,2W
1571 DATA 1S,2W,3S,10W,1S,2W,11S,2W,10S,1W,1C,1L
1580 DATA 6B,3P,5B,8I,13B,1I,2B,1I,2B,1I,17B,2S,3P,3B,1P,2B,1C,1S,1W
1581 DATA 1S,3W,3S,5W,8S,1W,7S,4W,10S,7W,1S,2W,1S,3W,10S,2W,1C,1L
1590 DATA 8B,4P,3B,7I,13B,1I,2B,1I,2B,1I,12B,1S,1P,3B,3P,8B,1C,2W,3S
1591 DATA 2W,1S,1W,4S,3W,7S,1W,1S,6W,3S,1W,2S,1W,2S,1W,4S,7W,1S,2W,2S
1592 DATA 1W,10S,1W,1C,1L
1600 DATA 15B,5P,2I,13B,1I,2B,1I,2B,1I,10B,2S,2P,2B,2P,9B,1C,3W,1S
1601 DATA 2W,1S,8W,7S,1W,10S,1W,2S,1W,2S,1W,4S,5W,2S,1W,9S,1W,2S,1W
1602 DATA 1S,2W,1C,1L
1610 DATA 14B,9P,5S,7B,1I,2B,1I,7P,5B,3S,1P,14B,1C,1W,1S,1W,2S,1W,2S
1611 DATA 6W,19S,1W,2S,2W,8S,1W,11S,7W,1S,1W,1C,1L
1620 DATA 12B,7P,18S,7P,5B,5S,1B,1P,2B,2I,9B,1C,2S,5W,1S,4W,42S,1W
1621 DATA 5S,4W,2S,1W,1C,1L
1630 DATA 12B,6P,3S,3P,6S,5P,3S,6P,3S,2P,2B,3I,7B,1I,2B,1S,1I,1C
1631 DATA 3S,5W,1S,3W,40S,3W,3S,4W,2S,1W,1C,1L
1640 DATA 13B,9P,5S,6P,6S,7P,2S,3P,2B,2P,2I,5B,2S,2I,1B,1I,1B,1I
1641 DATA 1C,3S,7W,1S,2W,38S,2W,4S,1W,2S,2W,6S,1W,1C,1L
1650 DATA 7B,1I,7B,15P,5S,4P,5I,2S,3P,3B,2P,2I,4B,3S,2I,3B,1I,1B,1C
1651 DATA 1S,1W,2S,3W,1S,3W,1S,3W,34S,3W,4S,1W,2S,1W,5S,1W,1S,1W,1C,1L
1660 DATA 8B,1I,8B,2P,5I,9P,7I,1B,6P,4B,2P,2I,4B,2S,3I,3B,2I,1B,1C,1S
1661 DATA 1W,3S,3W,1S,2W,3S,3W,23S,1W,6S,4W,4S,1W,2S,1W,6S,2W,1C,1L
1670 DATA 8B,1I,1B,1I,8B,21I,5B,2P,1B,1I,1B,2P,1I,5B,1S,2I,5B,2I,2B
1671 DATA 1C,3W,3S,2W,1S,1W,1S,1W,6S,1W,21S,5W,2S,1W,1S,1W,3S,2W,1S
1672 DATA 2W,3S,1W,3S,1W,1C,1L
1680 DATA 8B,4I,6B,27I,2B,2I,2P,5B,3S,1I,6B,1I,2B,1I,1C,2W,1S,1W,2S
1681 DATA 2W,5S,1W,2S,2W,27S,2W,13S,1W,3S,2W,1S,1W,1C,1L
1690 DATA 7B,7I,1B,37I,1P,5B,1S,1I,6B,1I,1B,2I,1C,2W,1S,2W,1S,1W,7S
1691 DATA 1W,48S,1W,3S,1W,1C,1L
1700 DATA 7B,44I,3B,1S,4I,8B,2I,1B,1C,1S,1W,2S,1W,55S,1W,1S,2W,2S
1701 DATA 1W,1C,1L
1710 DATA 8B,43I,1B,5I,4B,3I,3B,1I,1B,1I,1C,36S,1W,4S,2W,17S,1W
1711 DATA 4S,2W,1C,1L
1720 DATA 4B,3P,41I,2S,10B,7I,3B,1C,1W,13S,1W,21S,5W,12S,3W,1S,2W,5S
1721 DATA 1W,4S,1W,1C,1L
1730 DATA 9P,7B,30I,2S,10B,9I,3B,1C,9S,7W,12S,1W,2S,7W,13S,3W,1S,2W,7S
1731 DATA 1W,1C,1L
1740 DATA 10P,39I,5B,12I,4B,1C,14S,17W,6S,1W,1S,5W,7S,1W,1S,1W,7S,1W
1741 DATA 6S,2W,1C,1L
1750 DATA 10P,51I,7B,1P,1C,10S,6W,5S,1W,6S,1W,4S,6W,21S,2W,2S,1W,2S
1751 DATA 2W,1C,1L
1760 DATA 3I,7P,49I,10B,1C,14S,1W,1S,6W,3S,8W,26S,2W,2S,2W,1S,3W,1C,1L
1770 DATA 7I,10P,26I,2P,13I,10B,1S,1P,1C,21S,4W,33S,3W,2S,1W,1S,3W,1C
1771 DATA 1L
1780 DATA 9I,11P,15I,9P,3I,9P,1I,11B,1S,1P,1C,57S,3W,2S,3W,1S,2W,1C,1L
1790 DATA 8I,13P,14I,14P,4S,3P,1I,6B,4I,1S,2P,1C,57S,1W,1S,1W,1S,2W
1791 DATA 3S,1W,1C,1L
1800 DATA 3S,3P,2I,3P,3B,3I,3P,11I,3B,9P,9S,3P,2I,4B,6I,2P,1B,1C,59S
1801 DATA 2W,5S,1W,2S,1W,1C,1L
1810 DATA 2S,5P,2I,23B,13P,5S,4P,1I,3B,1P,7I,2P,2B,1C,59S,1W,9S,1W,1C
1811 DATA 1L
1820 DATA 5S,3P,22B,17P,2S,10P,3I,1P,3I,1P,2B,1I,1C,11S,6W,4S,5W,1C,1L
1830 DATA 5S,4P,20B,8P,7S,14P,1I,5P,1I,1P,2B,2I,1C,9S,20W,1C,1L
1840 DATA 2I,4S,4P,11I,12B,2P,13S,16P,2I,1B,3I,1C,10S,2W,14S,4W,1C
1841 DATA 1L
1850 DATA 8I,3P,13I,14B,11S,15P,6I,1C,11S,3W,15S,3W,1C,1L
1860 DATA 2I,4P,19I,18B,7S,13P,7I,1C,13S,3W,14S,6W,1C,1L
1870 DATA 7P,9I,14B,7P,9B,4S,13P,7I,1C,16S,3W,18S,6W,1C,1L
1880 DATA 5S,3P,2I,6P,2I,13B,8P,19B,5P,7I,1C,18S,3W,18S,2W,1C,1L
1890 DATA 6S,3I,4S,5P,12B,3I,7P,10B,4I,5B,4P,7I,1C,18S,5W,17S,4W,1C,1L
1900 DATA 20S,11B,5I,10P,10I,5B,3P,2I,1P,3I,1C,20S,4W,35S,2W,1C,1L
1910 DATA 21S,5B,12I,8P,2B,3I,3P,3I,6B,1P,1I,2P,3I,1C,21S,2W,37S,3W,1C
1911 DATA 1L
1920 DATA 3S,3P,10I,2P,2S,3P,6B,10I,3B,4P,3B,4I,3P,8I,6P,1C,23S,1W,37S
1921 DATA 3W,1C,1L
1930 DATA 3S,9P,15I,11B,2I,5P,5I,6P,5I,7P,2I,1C,23S,2W,22S,3W,9S,2W
1931 DATA 1C,1L
1940 DATA 1S,5P,4S,2P,1I,2S,3I,3P,5I,23B,7P,4I,10P,1C,23S,3W,32S,2W
1941 DATA 1C,1L
1950 DATA 3S,3P,1I,5P,15I,12B,4I,7B,4P,7I,9P,1C,24S,3W,31S,3W,1C,1L
1960 DATA 6P,2I,3P,10I,7B,18I,17B,7P,1C,23S,3W,32S,5W,1C,1L
1970 DATA 2I,3P,15I,9B,16I,5P,12B,8P,1C,23S,4W,31S,4W,1C,1L
1980 DATA 19I,13B,16I,7P,2I,6B,2P,2I,3P,1C,23S,2W,1S,1W,34S,2W,1C,1L
1990 DATA 1I,2B,9I,1B,2I,24B,4I,14P,4B,1P,5I,3P,1C,22S,1W,36S,2W,1C,1L
2000 DATA 6B,4I,17B,4I,4B,12P,4I,5P,14I,1C,20S,2W,34S,4W,3S,3W,1C,1L
2010 DATA 24B,2I,15B,7I,4P,15B,3I,1C,17S,7W,16S,1W,19S,1W,1S,5W,1C,1L
2020 DATA 23B,11I,10B,13I,9B,4I,1C,16S,5W,43S,2W,1C,1L
2030 DATA 23B,5I,5B,8I,4B,16I,3B,6I,1C,12S,6W,10S,1W,1C,1L
2040 DATA 36B,26I,3B,2I,3B,1C,11S,6W,1C,1L
2050 DATA 38B,25I,7B,1C,8S,7W,1C,1L
2060 DATA 39B,26I,5B,1C,8S,7W,1C,1L
2070 DATA 1E
END
```

If you enjoy driving, you're going to get a COMPUCRUISE. Once you see what it can do, you just won't be able to live without it.

This gadget fits into most dashboards . . . no strain even in a tiny sports car like the Mazda RX-7 . . . and once you have it, every trip is like flying a 747. The darned thing tells you the time, how fast you're going, how far you've been on this trip or since the last regassing, how many miles per gallon you're getting, either at the instant or the average on the trip . . . or gallons per hour at the moment or for the trip . . . temperature outside . . . inside (or coolant temperature, if you prefer) . . . oh, it has an elapsed time for the trip, a stop watch, lap time, an alarm . . . how much further for your trip, how many gallons more the trip will take, how much longer for the trip at your present average speed . . . yes, it gives you your average speed for the trip. You prefer it in metric, no strain . . . liters remaining, etc. Did we mention that it also has cruise control either at a speed set on the control board or at whatever speed you are traveling? The Compucruise will keep you busy and entertained during any trip . . . telling you more than you will ever want to know.

The Compucruise is not difficult to install . . . though it does connect to everything except the cigarette lighter. Until you've tried computerized travel, you haven't found out how much fun driving can be. It will work on any car not having fuel injection . . . and there is a front-wheel drive accessory gadget available for only $4.40-#P001 (regularly $5.50).

The price for the Compucruise is regularly $199.95 . . . and a bargain at that price. We'll sell you one of these fantastic gadgets for $159.95 with cruise control (Model 44-#P002), and $127.95 without (Model 41-#P003). Send money . . . and start having fun!

MOM'S

MAIL ORDER MICROS ✓53

Dept AKB • PO Box 427 • Marlboro NH 03455
Phone: (603) 924-3041

Steal Stopper

**Ever had your car stolen?
The first reaction is one of disbelief . . .
. . . you *know* it was *right there*!**

What you want is a modern combination lock on your ignition . . . The Steal Stopper. It's easy to install and almost impossible to defeat. *You* can by-pass it, if you want, for parking attendants or a car wash. Other than that, you set up a secret four digit code and only will then be able to start the car . . . even if you leave the keys in the ignition.

This protection retails for $50 . . . but we have a special for you at $39.95. Don't procrastinate. Order #P004.

Note: This product works best on Detroit cars. Mazda RX7 owners must order additional module, #P008, which costs $8. The Steal Stopper can be modified for Mercedes, Porsche, Ferrari, or other high performance European cars by returning unit to manufacturer with $3. They promise quick modification and return.

PROCESSOR TECH HARDWARE

7 **Processor Tech Video Display Module**—Memory mapped video for S-100, excellent condition. #D009-$144 each.

PROCESSOR TECH SOFTWARE

2 ★ **Processor Tech Extended Disk BASIC**—This is full disk BASIC on 8" disk for HELIOS II disk controllers with PTDOS and greater than 16K. #D015S-$70 each.

1 ★ **Extended Disk BASIC on cassette**—This is the same as previously mentioned for the Disk BASIC from Processor Tech. Needs more than 16K. #D017S-$62.

6 **Extended Cassette BASIC**—This includes all file operations, advanced functions for doing more than playing games; for SOLOS, CUTTER, and CONSOL Monitors. #D016-$22 each.

17 **BASIC 5 from Processor Tech**—This is a simple BASIC for a SOLOS, CUTTER, or CONSOL Monitor and 8K of RAM. #D013-$11.60 each.

4 **Processor Tech GAMEPAC** for above BASIC—Various simple games. #D014-$10.00 each

ASTROLOGICAL COMPUTER AND 4-FUNCTION CALCULATOR

ASTRO®

Gives you a fascinating look into your personality traits. Compares 16 different combinations. Matches your astrological influences to any day—past, present, or future. Gives in-depth analysis of your compatibility with your boss, your spouse, your lover, your child. Regularly $49.95. #P0020—$39.95 **SPECIAL PRICE.**

PROCESSOR TECH SOL

4 **SOL Computers**—8K RAM Monitor, S-100, excellent condition. #D004-$980 each.

17 **TREK-80** on cassette for SOL—This is one of the best real time space games available today; needs 8K. #D005-$11 each.

7 **Electric Pencil** on cassette for SOL—Word processor, needs 8K. #D006-$80 each.

MICRO TERM ACT TERMINAL

2 ★ **MicroTerm ACT Terminal**—Need a video monitor, up to 600 Baud, good condition. #S035S-$190 each. **SPECIAL PRICE.**

COMPUCOLOR HARDWARE

2 ★ **Compucolor Computer 8001** (use as computer or 75 MHZ Color Monitor)—8K RAM, BASIC and DOS in ROM, good condition. #S025S-$1400 each. **SPECIAL PRICE.**

2 ★ **Compucolor MiniFloppy**—5¼ inch, good condition. #S026S-$500 each. **SPECIAL PRICE.**

4 ★ **Compucolor 8K RAM card**—Static RAM, good condition. #S027S-$190 each. **SPECIAL PRICE.**

3 ★ **Compucolor Floppy Tape Drive**—Uses eight-track cartridges, good condition. #S028S-$60 each. **SPECIAL PRICE.**

PANASONIC TAPE DECKS

Panasonic RS261 US Stereo Cassette Decks—with auto-stop, record level adjust, VU meters, used condition; all have had heads replaced and aligned. #T001-$50.

Panasonic RS260 US Stereo Cassette Decks—same as above, but also has bias switch for chrome tapes. #T002-$50.

BRAND NEW* TRS-80 LEVEL II 16K
and accessories at FANTASTIC SAVINGS

LEVEL II 16K COMPLETE	$730	#TRS-001
16K EXPANSION UNIT	$425	#TRS-002
32K EXPANSION UNIT	$525	#TRS-003
DISK DRIVES	$425 each (Specify which drive you want)	#TRS-004
FRICTION MODEL PRINTER	$889	#TRS-005
TRACK MODEL PRINTER	$1350	#TRS-006

TERMS: Shipment normally within one week of receipt of your order (with cashier's check, money order, or credit card) for microcomputer and three weeks for accessories (checks take two weeks extra to clear bank). ADD $2.50 PER ITEM for HANDLING. Everything will be sent to you with UPS freight charges COLLECT.

**NOT UPGRADED USED OR RECONDITIONED LEVEL I's WITH OLD KEYBOARDS BUT BRAND SPANKING NEW TRS-80's IN FACTORY CARTONS WITH FULL FACTORY WARRANTY! COMPARE PRICES AND QUALITY AND ORDER FROM MOM's.*

MUSIC

1 **Software Technology S-100 Music system** on cassette—This is an S-100 Music system: contains the proper hardware. #D0058-$19.60 each.

ALS-80

14 **ALS-80 Operating system**—This system requires 12K RAM from D000 to FFFF; as well as either the SOLOS or CUTTER monitor; it includes an Assembler/Editor. #D018-$11.60 each.

NORTH STAR HARDWARE

1 ★ **North Star Floating Point BASIC card**—With special BASIC, new. #D0059S-$270.

2 ★ **North Star Floating Point BASIC card (kit)**—With special BASIC, S-100, new. #D0060S-$190.

7 **North Star Floppy Disk Controller card**—Single density, S-100, new. #D0061-$248.

92 **S-100 Edge Connector**—Gold Contacts, new. #D050-$2 each.

6 ★ **Extender Card for S-100 (kit)**—New. #D051S-$20 each.

COMPUTER TRAINER

1 ★ **IASIS COMPUTER-IN-A-BOOK**—8080 Microcomputer, comes built into training manual, excellent condition. #D020S-$225

ICOM DISK DRIVE ACCESSORIES

1 ★ **ICOM Dual Disk Drive**—Single density, 512K storage, S-100 controller, includes CP/M ROM, good condition. #S030S-$1400. **SPECIAL PRICE.**

1 ★ **ICOM PROM and 8" Disk for SOL FDOS**—This disk requires an ICOM S-100 Disk Controller installed in an S-100. #D031S-$150.

1 ★ **ICOM CP/M on 8" Disk for S-100**—Requires an ICOM S-100 controller in an S-100 cabinet, #D032S-$90.

2 ★ **ICOM FDOS-II on 8" Disk for S-100**—Requires an ICOM S-100 controller in an S-100 cabinet, no documentation. #D033S-$170 each.

1 ★ **ICOM FDOS-II on 5¼" Disk for S-100**—Requires an ICOM S-100 Mini-Floppy Controller in an S-100 cabinet. #D034S-$155.

MOM'S MAIL ORDER MICROS ✓53

Dept AKB • PO Box 427 • Marlboro NH 03455
Phone: (603) 924-3041

20% OFF Inventory Clearance SALE

SHUGART MINIDISK DRIVES

1 **Shugart MiniDisk Drives**—No cabinet, good condition. #S037-$300 each. **SPECIAL PRICE.**

IMSAI HARDWARE

1 ★ **IMSAI 8800 Mainframe S-100**—Excellent condition. #D0087S-$800.

1 ★ **IMSAI 80/15 S-100 Development System**—Partially assembled, needs a CPU card, excellent condition. #S0088S-$500 as is. **SPECIAL PRICE.**

1 ★ **IMSAI 80/15 S-100 Development System**—Kit, mainframe cover missing, needs a CPU card, excellent condition. #S0089S-$450 as is. **SPECIAL PRICE.**

1 **IMSAI 4K RAM card**—S-100, good condition. #D0055-$89.60

IMSAI SERIAL I/O CARDS

2 **IMSAI Serial I/O card 2-2 (kit)**—Two serial ports, full RS-232 control, S-100, new. #D0091-$124.

2 ★ **IMSAI Serial I/O card 2-1 (kit)**—One serial port, full control RS-232 control, S-100, new. #D0092S-$90.

IMSAI PARALLEL I/O CARDS

1 ★ **IMSAI Parallel I/O card 4-4**—Four parallel ports, S-100, excellent condition. #D0093S-$176.

1 ★ **IMSAI Parallel I/O card 4-1 (kit)**—One parallel port, S-100, new. #D0094S-$65.

IMSAI SOFTWARE

2 ★ **IMSAI IMDOS V2.02** on 8" Disk for S-100—No documentation, but this is apparently IMSAI's version of CP/M for S-100 systems with an IMSAI Disk Controller. #D0056S-$96 each.

1 ★ **TARBELL Cassette Interface (kit)**—Kansas City Interface, Tarbell Phase encoding, S-100, new. #D0064S-$80.

HEAD ALIGNMENT KIT

Best cassette recorder tape head alignment kit available. Solves loading problems. #K001-only $9.95.

PHONE INTERFACE

5 **Novation Modem #3102A**—Connects to any phone, originate only, good condition. #S021-$165 each. **SPECIAL PRICE.**

1 ★ **Novation Modem #43**—Connects to any phone, originate only, good condition. #S023S-$155. **SPECIAL PRICE.**

VECTOR GRAPHIC S-100

3 **Vector Graphic ROM/RAM card**—12K empty ROM sockets, 1K RAM, excellent condition. #D0078-$119.60.

5 **Vector Graphic Analog Interface**—Allows hobbyist to interface analog experiments, S-100, new. #D0079-$79.20.

INSTANT SOFTWARE HALF PRICE SPECIAL CLOSEOUT—ONLY $4

TRS-80, Level I, Games
Knights Quest/Robot Chase 4K-#ISI0003.
Cave Exploring 16K-#ISI0010.
Doodles & Display 16K-#ISI0030.
Fun Package #16K-#ISI0041.

TRS-80, Level I, Finance
Status of Homes 4K-#ISI0012.

TRS-80, Level II, Hobby
Model Rocketry Analyzer-#ISI0024.

PET
Personal Weight Control/Biorhythms-#ISI0006.
Digital Clock-#ISI0085.

TDL ZAPPLE

3 ★ **TDL ZAP 1K Monitor**—Simple monitor. #D0073S-$10.

BOOK CLEARANCE UP TO 50% OFF

Take a Chance with Your Calculator (Lithium-publisher) #BK1002—was $8.95, now $4.50.

Chemistry with a Computer (Educomp-publisher) #BK1010—was $9.95, now $5.00.

Computer Dictionary (Camelot-publisher) #BK1018—was $5.95, now $3.00.

FORTRAN Programming (Camelot-publisher) #BK1019—was $7.95, now $4.00.

FORTRAN Workbook (Camelot-publisher) #BK1020—was $4.95, now $2.50.

A Quick Look at BASIC (Camelot-publisher) #BK1043—was $4.95, now $2.50.

TDL SOFTWARE-DISK

1 ★ **TDL FDOS & SuperBASIC on 8" Disk**—This requires an ICOM Disk Controller and at least 20K of memory, plus a ZAPPLE Monitor in an S-100 Cabinet (Altair, IMSAI, etc.). #D0065S-$125.

1 ★ **TDL System Software on 5¼" disk**—This set of system software requires a North Star Disk Controller, and a TDL Systems Monitor Board I, and consists of 12K BASIC, Relocator/Linking Loader, Z-80 Editor, and Text Processor. #D0066S-$170.

1 ★ **TDL System Software on 5¼" disk**—This is the same as above, but does not require the Systems Monitor Board I. #D0067S-$190.

1 ★ **TDL System Software on 5¼" disk**—Again, as above, but requires a HELIOS Disk Controller and the TDL Systems Monitor Board II (not I). #D0068S-$170.

HONEYWELL

15 **Honeywell ASR-33 Communications Consoles** with TTY, paper tape reader and punch. Used, working when removed from service. Shipped freight collect or you pick up. Weight 300 lbs. $395. Order #P006.

MOUNTAIN HARDWARE

7 ★ **Mountain Hardware AC Controller**—Remote AC outlet control, S-100, new. #D040S-$90 each.

BALLY GAMES

2 ★ Bally VideoCode Cassettes—They consist of two games: Speed Math and Bingo Math. #D029S-$10 each.

HEURISTICS SPEECH LAB

1 ★ **Heuristics Speech Lab**—S-100, used, fair condition. #S042S-$90 as is. **SPECIAL PRICE.**

1 **Heuristics Speech Labs**—S-100, new. #D043-$151 each.

MORE FROM MOM'S ⇨

SPECIAL PRICE includes more than 20% discount. ★ indicates extra price reduction since last ad

Quantities are limited, immediate refund if ordered item is no longer available. *Phone answered by machine. Orders taken with credit cards. Questions answered by mail.

TERMS: FOB Marlboro, NH USA. Limited stock; everything guaranteed as described; you pay postage on returns. PRINT orders clearly. Minimum order $10 plus $2.50 shipping and handling charge in USA only. DOUBLE THAT ELSEWHERE. Orders over $50 add 5% for shipping in USA; 10% elsewhere (we will refund excess). Orders shipped UPS or insured mail only. No CODs please. Send US funds by check or money order. For credit card purchases, add 4%, list AE, MC or VISA, number, and expiration date. Mail to MOM's, Department AKB, PO Box 427, Marlboro NH 03455.

Condition of Inventory:
New = original container
Excellent = new, but not in original container
Good = tested or used in store

Qty	Catalog #	Description	Unit Price	Total

Delivery: 3 to 6 weeks. Personal checks take about 2 weeks to clear bank before we ship.
Enclosed $ ______

Bill: ☐ AE ☐ MC ☐ VISA

MOM'S MAIL ORDER MICROS ✓53
Dept AKB • PO Box 427 • Marlboro NH 03455
Phone: (603) 924-3041

Shipping & Handling ______
Credit Card (+4%) ______
Total ______

Card no. ______ Exp Date ______
Name ______
Address ______
City ______ State ______ Zip ______
Ship: ☐ UPS ☐ Insured mail Signature ______

✓Reader Service—see page 257

Time at a Glance on Your TRS-80

Your computer becomes a digital clock; it never sits idle.

Dave Rose
PO Box 20873
Atlanta GA 30320

If you are like most TRS-80 owners, you find yourself in front of your computer for no more than a few hours a day at most, whether it be for fun, business or both. How many times have you said to yourself, "Wouldn't it be nice if my TRS-80 could do something really useful during the time I'm *not* using it?"

Even if you've never asked yourself that question and even if you already *have* a digital clock (which this article describes), this program may be of interest to you. You may envision some intriguing possibilities in the ability to "paint" three-inch-high numbers on your Level II, 16K TRS-80 screen. That's exactly what this program does: it produces graphics for a digital clock that can be seen clearly from as far away as 100 yards! In addition to the clock routine, the program can also transform your TRS-80 into a giant-number digital counter.

The possibilities for shop and office adaptation here are obvious, especially in situations where many people can see your computer screen. When you're through computing for the day, simply load and run this program instead of turning the unit off. Voila! Now, instead of just sitting there absorbing skeptical thoughts from the office (or home) staff, your TRS-80 is suddenly providing a sure-to-be-appreciated service!

I use mine at home, where the computer is in the bedroom. With the brightness turned down, I have a combination night-light/clock, which is readable even to *my* sleepy eyes! And the nice thing is that my TRS-80 is now doing something useful whenever I'm not playing with it.

Once you understand the fairly simple methodology utilized in the program, there is no reason why you can't create similar programs of your own . . . programs that could produce letters, signs or any other figures on your TRS-80 screen, in any size you desire, and for any application.

The secret lies in the creation of "words" made up not of letters, but graphics symbols. Instead of consisting of the usual ASCII codes, these words are constructed (using the CHR$ and STRING$ functions) from graphics codes and certain control codes. A look at the program listing shows how this is done.

The Program

Steps 0–250 are your usual REM-type statements. As noted in the listing, I used many REM statements to help clarify the methods used (after all, most of the fun of any new program is what you learn from it!). However, you should delete these lines if you need the memory space.

Each of the numbers to be constructed consists of 188 graphics and control characters. Creating a string of this length using CHR$ and STRING$ statements is no simple task. You have to contend with either pages and pages of data statements or pages and pages of CHR$ statements. However, salvation in this case rests in one fact: most numbers are similar in makeup. For instance, the rounded bottom of the number 0 is also the bottom portion of the numbers 9, 6, 3, 5 and 8! So once several small building-block-type words are established, they can be concatenated into the 188-character monsters that are the three-inch-high numbers.

This obviously needs to be done before the clock mechanism is activated, because it does take up a lot of time and we don't want to goof up the timing routine. That's the reason for step 260.

Step 1900 has a dual purpose. It clears 3150 bytes of

Digital Clock program.

```
20 ' - PROGRAM CODED:
30 ' - ** RII - RADIO SHACK TRS-80 **
         ** LEVEL II ONLY, 16K **
35 '
40 ' - BYTE COUNT:          8288
45 '
50 ' - PROGRAM EXPANSION, VARIATION
       POSSIBILITIES, AND ANY OTHER
       QUESTIONS SHOULD BE REFFERED
       TO THE PROGRAM HARDCOPY
       DOCUMENTATION. . .

84 ' - CALIBRATION INSTRUCTIONS
        FOLLOW THIS LISTING. . .

90 ' #################################
     ########  WHAT  TO  DO  ########
     #### INSTEAD OF HITTING OFF ####
     #### DIGITAL CLOCK PROGRAM  ####
     ########  BY DAVE ROSE  ########
     #### BOX 20873, ATLANTA, GA ####
100 '########### 30320 ##############

110 '   THIS PROGRAM PRODUCES A DIGITAL
        CLOCK WITH 3-INCH-HIGH NUMBERS.
        EACH NUMBER IS ACTUALLY A "WORD"
        MADE UP OF 188 GRAPHICS AND CON-
        TROL CHARACTERS.

120 '   THE PROGRAM LISTING HAS BEEN
        LIBERALLY SPRINKLED WITH "REM"
        STATEMENTS TO AID YOU IN UNDER-
        STANDING HOW THIS WAS ACCOMP-
        LISHED.  FEEL FREE TO DELETE
        THESE STATEMENTS IF YOUR OWN
130 '   MEMORY NEEDS SO DICTATE. . .

        IT TAKES A LONG TIME TO "BUILD"
        THOSE GRAPHICS WORDS, SO WE DO
        THAT BEFORE WE SET THE CLOCK. . .
260 CLS: PRINT@ 399,
    "THIS WILL ONLY TAKE 9.87 SECONDS"
```

Giant digital clock.

memory for string manipulation and also uses the DEFSTR function to define those variables beginning with B, C or P as string variables. This function is extremely valuable in any program that requires a lot of string handling. Not only does it save wear and tear on the $ key, but, more important, it frees a lot of memory that would otherwise be used up by those dollar signs.

Steps 1910–2320 create the smaller building-block words I mentioned earlier (the top and bottom of 0, the middle of 3, etc.).

Steps 3000–3540 use these building blocks to construct the actual numbers, which consist of 14 graphics characters, followed by 15 control characters, followed by 14 graphics characters, etc., seven times over. The control characters serve to back the cursor up 14 spaces, then drop it down to the next line. The end result is a number 14 characters wide and *seven* lines tall. Some of the numerals (the seven, for instance) are unique, and so must be constructed practically one character at a time.

Notice that the numeral-words are named C(0) through C(9). The convention of using the same subscript number as the number that the word depicts eases the task of calling these words from memory for display purposes. Notice also that the smaller, building-block words are erased from memory (such as in steps 3190, 3230) once their job is completed. This frees memory space that would otherwise be uselessly tied up.

Steps 3600–4100 contain the clock routine. Of interest here is the delay subroutine at step 4100. At first it seems to be a useless subroutine, until you notice that the program is using its own execution time as part of the timing mechanism. When it's time for a number to change, the new number is painted over the old one. This causes the loop to execute much more slowly on that particular pass. A delay routine then becomes necessary to smooth out the loop execution time.

You might say, "Why don't you just repaint the entire display with each iteration of the loop?" Well, actually that works pretty well, except that the display is then filled with annoying tracer marks as the cursor goes zooming through the screen. It looks like bad reception on a black-and-white TV set.

Calibration of the clock for your particular machine will undoubtedly be necessary. It seems that all TRS-80s are *not* created equal! I tried the program out on three different units, and although it ran beautifully, the clock had to be recalibrated in each case. The details for calibrating the clock

```
1900 CLEAR 3150: DEFSTR B,C,P
1910 DIM B(12), P(15)
1920 '  WE BEGIN BY FORMING SMALL
        GRAPHICS "WORDS" WHICH CAN BE
        COMBINED TO FORM THE NUMBERS.

1930 '  FOR INSTANCE, STEPS 2020  TO
        2050 CREATE THE ROUNDED TOP
        FOR THE 2, 3, 6, 8, 9, AND 0.

1960 FOR X=1 TO 12: B(X)=STRING$(X,32):
     NEXT X
1970 FOR X=1 TO 15: P(X)=STRING$(X,191):
     NEXT X
2000 CB=STRING$(14,24)+CHR$(26)
2020 CZ=CHR$(160)+CHR$(188)+P(10)
       +CHR$(188)+CHR$(144)+CB
2040 CX=P(3)+CHR$(135)+B(6)+CHR$(139)
       +P(3)+CB
2050 CC=CZ+CX
2060 CD=P(3)+B(8)+P(3)+CB
2080 CY=P(3)+CHR$(180)+B(6)+CHR$(184)
       +P(3)+CB
2100 CW=CHR$(130)+CHR$(143)+P(10)
       +CHR$(143)+CHR$(129)+CB
2110 CE=CY+CW: CF=P(14)+CB
2140 CG=B(11)+P(3)+CB
2160 CH=P(3)+B(11)+CB
2180 CI=B(6)+P(6)+CHR$(183)+CHR$(145)
       +CB
2200 CJ=B(2)+STRING$(8,176)+CHR$(190)
       +P(2)+CHR$(159)+CB
2220 CK=CHR$(184)+P(3)+STRING$(8,143)
       +CHR$(131)+B(1)+CB
2240 CL=P(3)+CHR$(149)+B(10)+CB
2260 CM=CHR$(176)+CHR$(188)+P(1)
2280 CN=CHR$(176)+CHR$(188)+P(4)
2300 CO=B(6)+P(3)+B(5)+CB
2320 CP=CHR$(176)+CHR$(188)+P(2)
       +CHR$(143)+CHR$(131)

3000 '
      #### NOW WE BUILD A ZERO

3020 C(0)=CC+CD+CD+CD+CE: CD=""
3040 '
      #### THEN AN EIGHT

3060 C(8)=CC+CE+CHR$(27)+CC+CE
3080 '
      #### A SIX

3100 C(6)=CC+CH+LEFT$(CF,4)
        +RIGHT$(CZ,25)+CX+CE
3120 '
     #### A NINE
3140 C(9)=CC+CY+LEFT$(CW,5)
        +RIGHT$(CF,24)+CG+CE:
        CW=""
3160 '
     #### A FIVE
3180 C(5)=CF+CH+LEFT$(CF,4)
        +RIGHT$(CZ,25)+B(4)
        +RIGHT$(CX,25)+CG+CE
3190 CZ="": CH=""
3200 '
     #### A THREE
3220 C(3)=CC+B(4)+RIGHT$(CY,25)
        +CI+B(4)+RIGHT$(CX,25)
        +CE
3230 CY="": CX="": CE="": CI=""
3240 '
     #### A TWO
3260 C(2)=CC+CG+CJ+CK+CL+CF+CB
3270 CC="":CG="":CJ="":CX="":CL=""
3280 '
     #### A ONE
3300 C(1)=B(6)+CM+B(5)+CB+B(3)+CN
        +B(5)+CB
3310 C(1)=C(1)+CO+CO+CO+CO+B(3)
        +P(9)+B(2)
3320 '
     #### A FOUR
3340 C(4)=B(9)+CM+B(2)+CB+B(6)
```

```
          +CN+B(2)+CB
3350 C(4)=C(4)+B(3)+LEFT$(CP,5)
          +P(4)+B(2)+CB
3355 C(4)=C(4)+CHR$(160)+RIGHT$(CP,5)
          +B(2)+P(4)+B(2)+CB
3360 C(4)=C(4)+P(3)+STRING$(5,188)
          +P(4)+STRING$(2,188)+CB
          +STRING$(8,131)+P(4)
          +STRING$(2,131)+CB+B(8)
          +P(4)+B(2)
3380 '
     ####  A SEVEN (FROM SCRATCH)
3400 C(7)=P(14)+CB+B(10)+CHR$(184)
          +P(2)+CHR$(159)+CB
3420 C(7)=C(7)+B(7)+CP+B(1)+CB+B(4)
          +CP+B(4)+CB
3440 C(7)=C(7)+B(1)+CHR$(160)+CHR$(184)
          +P(2)+CHR$(159)+CHR$(135)
          +B(7)+CB
3460 C(7)=C(7)+CHR$(168)+P(2)+CHR$(159)
          +CHR$(129)+B(9)+CB
3480 C(7)=C(7)+CHR$(170)+P(2)+CHR$(149)
          +B(11)
3500 FOR K=1 TO 12: B(K)="": NEXT K
3520 FOR K=1 TO 15: P(K)="": NEXT K
3540 CB="": CP=""
3550 CLS: PRINT STRING$(3,26);
     "      1. DIGITAL CLOCK"
3560 PRINT "      2. DIGITAL COUNTER"
3580 INPUT"WHICH DO YOU WANT";QQ
3590 ON QQ GOTO 3600, 4490
3600 PRINT@383," ":
     PRINT"     NOW SET THE CLOCK..."
3640 PRINT"     WHAT IS THE CURRENT:"
3670 PRINT
3680 INPUT"                   HOUR";TE
3700 TF=INT(TE/10): TE=TE-(TF*10)
3720 INPUT"                MINUTES";TC
3730 TD=INT(TC/10): TC=TC-(TD*10)
3760 INPUT"                SECONDS";TA
3770 TB=INT(TA/10): TA=TA-(TB*10)
3780 CLS: CQ=CHR$(170)+CHR$(191)
          +CHR$(149)
3790 PRINT@478,CQ; STRING$(3,24);
          STRING$(2,26);CQ
3795 ZD=77: ZF=77: ZE=77: ZC=77
3800 FORN=1TO77:NEXTN:TA=TA+1
3820 IF TA<=9 GOTO 4000
       ELSE TA=0: TB=TB+1
3840 IF TB<=5 GOTO 4000
       ELSE TB=0: TC=TC+1
3860 IF TC<=9 GOTO 4000
       ELSE TC=0: TD=TD+1
3880 IF TD<=5 GOTO 4000
       ELSE TD=0: TE=TE+1
3900 IF TE>9 THEN TE=0: TF=TF+1
3920 IF TF=1 AND TE=3 THEN TA=0:
       TB=0: TC=0: TD=0: TE=1: TF=0
4000 IF TF<>ZF THEN PRINT@320, C(TF)
       ELSE GOSUB 4100
4010 IF TE<>ZE THEN PRINT@335, C(TE)
       ELSE GOSUB 4100
4020 IF TD<>ZD THEN PRINT@354, C(TD)
       ELSE GOSUB 4100
4030 IF TC<>ZC THEN PRINT@369, C(TC)
       ELSE GOSUB 4100
4040 PRINT@ 055, TB;TA
4050 ZC=TC:ZD=TD:ZE=TE:ZF=TF:GOTO 3800
4090 END
4100 FOR X=1 TO 42: NEXT: RETURN
4490 CLS: PRINT@320,
     "PRESS ANY KEY TO ACTIVATE"
4495 FOR X=1 TO 1000: NEXT: CLS
4500 CS=INKEY$: IF CS="" GOTO 4500
                  ELSE TH=TH+1
4520 IF TH<=9 GOTO 4600
       ELSE TH=0: TI=TI+1
4540 IF TI<=9 GOTO 4600
       ELSE TI=0: TJ=TJ+1
4560 IF TJ<=9 GOTO 4600
       ELSE TJ=0: TK=TK+1
4580 IF TK<=9 GOTO 4600
```

Clock as a night-light.

routine have been made a part of the program listing so that even if you lose this article, you'll still be able to use the program effectively.

Steps 4490–4650 contain the counter routine. This is basically just a single-iteration counter. The numbers are displayed just as in the clock routine. The INKEY$ function is utilized here so that hitting *any* key will cause the counter to increment. This routine was included primarily to give you a starting place for tailoring these enlarged display techniques to your own needs.

Program Expansion Ideas

I've noted here some ideas for possible alteration and/or expansion of this program. Most are simple and involve only minimal changes to the program structure.

1. A digital timer.
2. An alarm clock (this one is easy if you have a printer with a software-accessible bell or tone).
3. A digital measuring device (using the out and inp functions).
4. A counter that increments the display by different amounts according to which key is struck (the keys 1 through 9, for instance, could cause the display to advance their respective amounts).
5. A device that would perform different math operations on the number in the display, according to which key is struck (this application has awesome classroom possibilities).

That should be enough to get the imagination pumping! By the way, if you are one of those lucky folks who are blessed with a disk system, then I have especially good news for you. You can use the real-time clock that is part of your disk operating system as your clock routine! This exempts you from any calibration problems whatsoever. All you need do is make a couple of small changes in the program: delete lines 3600–4100 and insert a routine to compare what's in the display to the DOS TIME$ string and ensure that they are the same. Also, be sure that the TIME$ string is set to the correct time before running the program. ■

```
       ELSE TK=0: TL=TL+1
4600 IF TL<>0 THEN TL=0: TK=0:
       TJ=0: TI=0: TH=0
4620 PRINT@320, C(TK):
     PRINT@336, C(TJ)
4640 PRINT@352, C(TI):
     PRINT@368, C(TH)
4660 GOTO 4500
4680 ' ######################
4700 ' TO CALIBRATE THE CLOCK:
4720 '  ALL CALIBRATION IS DONE
        BY EDITING STEP NUMBER
        3800.  USE THE FOLLOWING
        METHOD:
4740 '  1. INITIAL CALIBRATION-
          ADJUST THE ITERATION
          NUMBER WITHIN THE FOR-
          NEXT LOOP
4760 '  2. FINE TUNING-
          ADJUST THE NUMBER OF
          SPACES WITHIN THE FOR-
          NEXT LOOP
4780 '  3. EXTRA FINE TUNING-
          ADJUST THE NUMBER OF
          SPACES BEHIND THE
          SECOND COLON
4800 END
```

Instant Software: BASIC Sophistication

Every once in a while you come across a piece of software that makes you stop and say to yourself, "That's the way it's supposed to be done." This quintet of packages is just that kind of elegant and efficient software.

On this page, we feature: an Energy Audit package for homeowners *and* the Accounts Receivable/Accounts Payable system. Both of these TRS-80 programs feature the speed and convenience of floppy disks.

On the next page, three cassette-based packages: our Music Master—for those interested in computer music and our two Utility packages. Whichever you may choose, our attention to detail and dedication to producing user-oriented programs means you'll always be satisfied with ISI.

ENERGY AUDIT

When it comes to spending money for energy to heat a home, everybody's a "skinflint." With today's rising fuel prices and staggering inflation, you need ways not only to cut down on your home heating bills but also to conserve our diminishing energy supplies.

One of the best ways to invest your money is to insulate your home. You'll save money, and your home will appreciate in value. But, you're asking yourself, "How do I start, where should I insulate, and what will it cost me?" The Energy Audit package can answer all these questions and more.

The programs in this package will give you detailed instructions on how to examine your home, collect information on its structural components and layout, and create a data file. The program will then analyze this data, show where you're losing money because of excessive heat loss, and describe the most cost-effective methods for saving energy and money.

You may select any area of your home for detailed scrutiny, input your own cost figures, calculate savings potentials for different options, and incorporate whichever you want into your own home-investment budget. In effect, you can create a computerized "model" of your home. You'll be able to see the changes that will be made in your energy budget as a result of adding a wood stove or refinishing an attic.

When your friends and neighbors see how much money you're saving, they will want their homes analyzed too. You could even start your own business, showing your customers exactly where and how to insulate their homes for maximum savings. The Energy Audit package is the perfect selling tool for any contractor or building supplies dealer.

With the Energy Audit package you'll know what your home needs, where it needs it, and how much it will cost you.

Or maybe you enjoy paying utility bills?

Pkg. 0089R (cassette version) $49.95.
Pkg. 0052RD (disk-based version) $75.00.

AVAILABLE ON FLOPPY DISKS

ACCOUNTS RECEIVABLE/ ACCOUNTS PAYABLE

Now, in one package, you can have a *complete* Accounts Receivable/Accounts Payable (AR/AP) system! These programs will handle all the drudgery involved in processing AR/AP entries.

Each program is capable of handling up to 760 accounts and as many as 1500 entries per month. Should you need to handle more accounts, you can divide them into as many sub-groups as necessary and keep a set of data disks for each sub-group.

The Accounts Receivable program can print invoices, statements, and address labels for each customer. It will also generate a Month End report, a Customer Activity report, and a Daily Sales report.

The Accounts Payable program will generate a Month End report, an Account report, a Daily Activity report and a Check Payment report.

The AR/AP package is ideal for any small business. The programs are self-prompting and are easily used by anyone familiar with AR/AP operations.

These programs can save you money, because they can print your company's letterhead at the top of each invoice and statement, using plain, fan-folded paper.

Accounts Receivable/Accounts Payable: Software for the Professional.

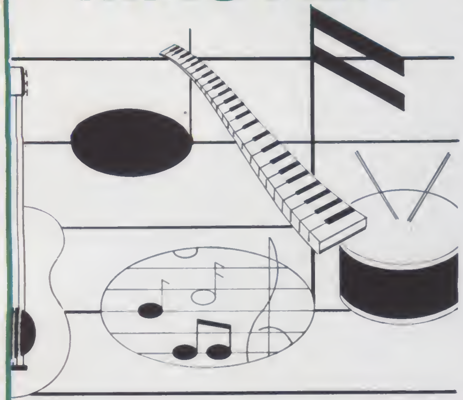
THIS PACKAGE REQUIRES THE FOLLOWING MINIMUM SYSTEM

1. A TRS-80 Level II microcomputer, 16K of memory.
2. An Expansion Interface with at least 16K of additional memory.
3. Three mini-disk drives.
4. A pin-feed line printer.
5. Any TRS-80 Disk Operating System.

Ask for Package 0075RD **$199.95.**

Instant Software Inc.™ Peterborough, New Hampshire 03458 603-924-7296

MUSIC MASTER



Breathes there a programmer with soul so dead
who never to himself has said,
"I wish I could get music out of a TRS-80!"

Now you can. Music Master lets you compose music, play your keyboard as if it were a piano, and experiment with programming to produce music suited to your taste. This package includes:

•**MICRO ORGAN**—Change your computer into a musical instrument. You'll have a range of four octaves with three voices. The program will let you play flats and sharps to imitate the sounds of an organ, a harpsichord, or a piano.

•**KALEIDOPY**—Now you can have a computerized "player piano." Generate a symmetrical graphics pattern and then see it transformed into music.

•**COMPOSER**—Experiment with computer-generated music. This program allows you to select the length of the piece, the scale it will be played in, and the tempo. The instructions even include a "messing-around" section that tells you how to create your own special effects.

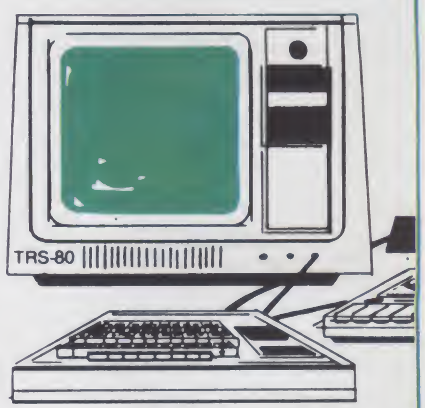
•**KEYMANIA**—This game will test not only your memory, but your musical ear. From one to four players can try to remember and repeat the melody the computer creates. You can select the number and range of notes in the tune, the duration of each note, and the intervals between notes.

They may laugh when you sit down at the keyboard of your computer, but not after they hear what the Music Master package can do.

(This package uses an optional audio amplifier, e.g. Radio Shack # 277-1008).
For the TRS-80 Level II 16K. **Order No. 0084R. $7.95.**

For a free catalog listing over 200 programs write:
Instant Software Catalog Dept., Peterborough, N.H. 03458.

UTILITIES



TRS-80 UTILITY I Ever wonder how some programmers give their programs that professional look? Instant Software has the answer with the TRS-80 Utility I package. Included are:

•**RENUM**—Now you can easily renumber any Level II program to make room for modification or to clean up the listing. (Only for use with 16K of memory.)

•**DUPLIK**—This program will let you duplicate any BASIC, assembler, or machine-language program, verify the data, and even copy Level I programs on a Level II machine.

For the TRS-80 Level II 16K. **Order No. 0081R $7.95.**

TRS-80 UTILITY II Let Instant Software change the drudgery of editing your programs into a quick, easy job. Included in this package are:

•**CFETCH**—You'll be able to merge BASIC programs, with consecutive line numbers, into one program. CFETCH can also search through any Level II program tape and display the file names for all the programs.

•**CWRITE**—Combine subroutines that work in different memory locations into one program. CWRITE works with BASIC and/or one or more machine-language programs. It will even give you a general checksum to verify that your program hasn't dropped any bits.

Another fine tool for your TRS-80 Level II 16K. **Order No. 0076R $7.95.**

If the store nearest you does not stock Instant Software, use this order blank to purchase your software directly, or call Toll-Free 1-800-258-5473.

Order Your Instant Software today!

Name ______________________

Address ______________________

City ______________ State ______ Zip ______

☐ Check ☐ Money Order ☐ VISA ☐ master charge ☐ American Express

Credit card# ______________ Exp. date ______

Signature ______________ date ______

Quantity	Order No.	Program name	Unit Cost	Total cost
		Shipping		$1.00
			Total order	

Mail to:
Instant Software™
Peterborough NH 03458 USA Dept. 40D0

Prices Valid In USA Only

*A trademark of Tandy Corporation

Instant Software Inc. Peterborough, New Hampshire 03458 603-924-7296

APPLE II BOWLING SECRETARY

Calculates all of the following items:
For individual—name, team#, handicap, running total pins, number of games bowled, average, no. of 200'S, no. of 500 series, no. of 600+ series, hi game (scratch), hi series (scratch).
For team—team#, team name, wins, losses, pct., team total pins (scratch), team avg., team hi game, team hi series.
All simply by entering 3 weekly scores per bowler! Also maintains subs records and allows for blind scores. Provides for weekly reporting with printer option.
(Disk and 48K required) $24.95
Full line of Programma and Muse software available. To introduce you to Mighty Byte take 15% off any order.

MIGHTY BYTE COMPUTER INC.
P.O. Box 213
HO-HO-KUS, NEW JERSEY 07423
(201) 445-8256 ✓205
VISA AND MASTERCHARGE ACCEPTED.

COMPUCOLOR INTECOLOR

SALES • SERVICE • SOFTWARE
BUY • SELL • UPGRADE
WHOLESALE—RETAIL

UPGRADE—your intecolor or compucolor to the latest configuration.

SERVICE—providing local service and service on the west coast for east coast firms.

INTERNATIONAL—users group.

SOFTWARE—over 700 programs in our library. Join to purchase low cost software.

PROGRAM EXCHANGE—5 to 10 programs for each acceptable submission.

DATA EXCHANGE—printer interfaces, S100 bus, new op systems, etc.

CLUB DISCOUNTS—on compucolor software and hardware.

MEMBERSHIP BULLETINS—1 year—$30.00

FOREIGN ORDERS—add $10.00, add $16.00 if check is not on a U.S. bank.

FORMATTED DISKS—$4 each—10 minimum—$2 per order postage.

S. P. ELECTRONIC SYSTEMS
5250 Van Nuys Blvd.
Van Nuys CA 91401
Phone 213/788-8850
✓143
MON-FRI 10AM-6PM
SAT 12-3PM

Available Direct From DI$COUNT COMPUTER PRODUCT$

Master Charge & Visa Accepted
(Please give card no. and expiration date)

APPLE		**Texas Instruments**	
APPLE II 16K	$999.	Model 810	
Micro Music		Basic Printer	1649.
Board for Apple	168.	**Centronics**	
OHIO SCIENTIFIC		779-2	1049.
SUPERBOARD II	259.	Tractor Feed Printer	
C1P 4K	329.	Epson Tx-80	
C1P 8K	359.	Tractor Printer	688.
C4P 8K	598.	**Comprint Computer Printer**	
C4P MF	1498.	**International**	
C8P	795.	Comprint 912 S	598.
C8P DF	2349.	**Hazeltine**	
C2 OEM	2399.	1410	798.
Novation Inc.		1420	898.
Cat Modem	179.	1500	998.

MAIL & PHONE ORDERS ONLY!
SHIPPING EXTRA
DELIVERY FROM STOCK TO 6 WEEKS
P.O. BOX 308 ✓199
Thiells, N.Y. 10984 **(914) 429-9631**

IN NEW YORK CITY

...Look to Aristo/Polks for reliability, quality, product versatility and value...

OHIO SCIENTIFIC & ARISTO/POLKS FULL STOCK AND SERVICE ON CHALLENGER MICRO-COMPUTERS.

CHALLENGER C1P 8K	$399.00
C1P 5" FLOPPY 20K	$1198.00
SUPERBOARD C1P	$279.00
CHALLENGER (COLOR) C4P 8K	$698.00
C4P FLOPPY 24K 5"	$1698.00
CHALLENGER C8P	$895.00
COLOR-DUAL 8" FLOPPY C8P	$2597.00
C-3 48K DUAL FLOPPY 8"	$3995.00
C-3 + 23 MEG HARD DISK	$9900.00
C-2-OEM 48K DUAL FLOPPY 8"	$2699.00

PLUS ALL SOFTWARE & PERIPHERALS

Mail order invited if machine can be sent back to us for service. 220V conversions available for systems-write for quote. Write for free catalog. *M C. VISA. AX CARDS ACCEPTED* ✓220

Aristo/Polks • 212-279-9034
314 5TH AVE.(32 ST) N.Y.C., N.Y. 10001
9 30/6-DAILY ■ THURS TIL 9 ■ SUNDAY 11-5

...Look to Aristo/Polks for reliability, quality, product versatility and value...

PROGRAM & RUN EPROMS

The MORE board plugs on to a

KIM, SYM or AIM

this union makes a potent EPROM programmer/runner.
FEATURES INCLUDE: Sockets for 3K of 2114 RAM expansion • zero force sockets for both burn/verify and run EPROMS • includes all EPROM personality keys to allow run, copy, burn and/or verify for 2708, 2716 (±5, +12V) and 2716, 2758, TMS 2516 (5V only) • 16 bit output port with LED on each bit • -5 and +26V regulated supplys • MORE board requires only +5V and +12V at 200 ma • address any K on 8K boundries • extends KIM'S edge connectors • P.C. board is plated through double sided epoxy • **Manual, MORE board, all personality keys, software on tape (KIM format) and listings** -- $169.95 • Extra -- 2708 EPROM with software $20.00

✓204

T.T.I. **P.O. Box 2328** **Cookeville, TN 38501**
Phone: 615-526-7579

16 K UPGRADE

$66.95

TRS80, APPLE II, AND SORCERER

HIGH QUALITY PRIME 16K RAMS FOR MEMORY UPGRADE. THE KIT INCLUDES, THE RAMS, SHUNTS AND INSTRUCTIONS TO ALLOW EASY UPGRADE IN MINUTES. ALL PARTS CARRY 12 MONTH WARRANTY.

TO ORDER, SPECIFY KIT AND ENCLOSE CHECK OR MONEY ORDER. ADD $2.00 POST AND PACKING; TEXAS RESIDENTS ADD 5% SALES TAX.

IAN ELECTRONICS
P.O. BOX 14079 ✓209
AUSTIN, TEXAS 78761

TRS-80 LEVEL 1 4K EDUCATIONAL SOFTWARE

ELECTRONIC SERIES:

6 programs plus graphics covers resistance Ohm's law capacitance capacitive reactance resonant circuits energy and power Cassette No 123179KP $9.95

METRIC SERIES:

6 programs for consumers Each program includes computer generated tests Covers prefixes length, weight, volume, and temperature
Cassette No 122579JI $9.95

Let us market **YOUR** programs. Either royalty or direct puchase. Send S.A.S.E. for details.

Include $1.55 for postage and handling Send check M.O. to

NORKA SOFTWARE
BOX 6203
AKRON, OH 44312 ✓212

NUDES

BARE BOARDS? What is the use of the Best Bare Board/Kit, I/O Board if no one knows about it? This board has everything! 48 lines of parallel I/O, completely programable including hand shaking and bidirectional ports. The serial I/O is programable for asyncronous/syncronous/bisyncronous communication. Included in this Board is probably one of the most useful and underated I/O components available, a counter/timer chip. This chip can be used as a programable baud rate generator, a real time clock, a counter, a programable one shot, a programable divide by N counter, and several other uses besides!! Incredible as all this may seem, We are not through yet! This Board has 1K of RAM space and 3K EPROM space. All this on One Board! The Master I/O Board is $39.95 plus $2 shipping & handling; the Complete Kit including one 2708 EPROM is $138.95 plus $2 shipping & handling. For $49.95 additional we will send you the complete Zapple Z-80 Monitor, one of the best debug monitors available, customized for the Master I/O Board including 3 EPROMS and documentation.

Order from R.W. Electronics, 3165 N. Clybourn, Chicago, IL 60618, 312/248-2480. Master Charge & Visa welcome. We ship worldwide. Call or Send for Catalog. Circle reader service card no. ✓210

Kit $138.95 master charge VISA **Board $39.95**

KEYBOARD EXPANDOR FOR APPLE II*

C&H Micro announces the transformation of the APPLE II into a complete upper and lower case system. KEYBOARD EXPANDOR, a hardware-software modification of the APPLE II, actually allows the shift keys to be used just like a conventional typewriter.

The hardware change is a one-wire modification requiring one solder point. The software is a 1/4K transparent machine language routine which augments the Monitor. All APPLE characters and editing keys maintained. Cap and Shift Locks and an Inverse mode display option included. Compatible with methods displaying ASCII such as Paymar's LCA and the HI RES CHARACTER GENERATOR. Totally compatible with DOS, allowing use of U/L in TEXT files, PRINT and REM statements, DOS file names, and Immediate mode.

Full documentation. Software provided on disk. Orders, with certified check or money order for $20.00, should be sent to:

C&H Micro
P.O. Box 249
Clifton Park, NY 12065

**Apple is a registered trademark of Apple Computer Inc.*

Do-All-Plus

The original Do-All was a winner. This CBASIC conversion is even better.

Thomas E. Doyle
5222 Big Bow Rd.
Madison WI 53711

One of my duties as treasurer of the Madison Area Repeater Association (MARA, Inc.) is to prepare financial reports for publication in our newsletter. MARA is an amateur radio club that owns and operates an amateur radio repeater system. As the club grew from a membership of 12 to its present membership of over 200, the time required to prepare these reports increased considerably.

The newsletter is published four times each year, and each quarter I would look at my computer and think how much easier the job would be if the club records were on the computer. After considering the time required to write the programs to handle the tasks, I would get out the calculator, paper and pencil and prepare the report by hand. A decision by the club to publish the newsletter eight times each year provided the necessary incentive for me to write the programs.

Writing Applications Programs

I believe the first step in writing good applications programs is to first forget about the computer and analyze the tasks as they are presently being done by hand. I reviewed the club books to categorize the nature of income and expenditures. I identified four income categories:

1) Dues paid by each member on a yearly basis
2) Fund-raising activity—Swapfest
3) Sale of club property
4) Donations to the club

I discovered eleven expense categories:

1) Tower space rental for repeater antennas
2) Newsletter printing and mailing
3) Phone bills
4) Social—meeting refreshments, etc.
5) Fund-raising activity—Swapfest
6) Donations to other groups—Red Cross, etc.
7) Post office box rent
8) Equipment purchases
9) Repeater antenna expenses
10) Printing—membership cards, etc.
11) Misc. debits—checking-account charges, etc.

The existing financial reports consisted of a breakdown of club income and expenditures by category on a quarterly basis. In addition to preparing the financial reports, tasks include entering and correcting items, sorting items by date or category and printing lists of items.

After determining the tasks to be performed, it is necessary to consider the available hardware and structure of the data file. For this application, information on each income or expense item will be stored on a disk file. One record on the file will be allocated to each item. A record will contain four pieces of information on the item:

1) Transaction code, which will identify the item by category
2) Amount of income or expense
3) Description of item
4) Date on which the income or expense occurred

The income and expense items will be stored together in the same file. The transaction code will identify the nature of the item as an income or an expense. To allow for possible new income categories, ten transaction codes were allocated for income categories. Since the first ten transaction codes were reserved for income items and the remaining transaction codes are reserved for expense items, programs working on this file can easily determine the nature of an item as an income or expense.

The next step in writing good applications programs is to search the literature for programs written by others to solve similar problems. Studying other programs is time well spent and may prevent the programmer from reinventing the wheel. A literature search uncovered one program that would offer an excellent basis for solving this problem: the Do-All program by Randy Miller (*Kilobaud*, August 1977, p. 84). The Shell-Metzner sorting routine modifications ("5 Minutes or 5 Hours?" *Kilobaud*, May 1978, p. 100) were added, and the program was converted to CBASIC and tried out.

Program Modifications

There were a few problems in using the Do-All program in this application. Rather than rewrite the entire program, I decided to modify it.

1) Add disk I/O, since the original program was set up for cassette.

2) Store transactions by code number rather than description.

3) Sort items by data, even if the items span more than one year.

4) Modify the print function to include breaks between pages.

There are two ways in which the disk I/O modification could be implemented. The structure of the Do-All program could be left relatively intact, which would require that the entire data file be read into system memory from the disk. Operations on the file, such as adding items, changing items and sort-

ing, would be carried out by making changes to the file while it is in system memory.

After the changes to the file are made, the entire data file would be copied from memory to the disk. This method operates at high speed and minimizes wear and tear on the disk system. The disadvantage of this method is that it requires a large system-memory area.

Another way of adding disk I/O is to bring data from the disk file into system memory in small sections. This method has the advantage that it requires only a small amount of the system memory but it operates at a much slower speed in non-sequential operations such as sorting. The amount of RAM in the system (48K) allowed over 200 items to be in system memory at once. This was adequate to hold more than one year's data, so I chose the first method of modifying the program.

I added disk I/O to the Do-All program by modifying the LOAD and DUMP routines. If the LOAD function is selected, the program asks if the data file is to be loaded from the terminal or from the disk. If terminal input is specified, the Do-All program continues normally. If disk input is specified, the program jumps to a routine (starts at line 11000) to load a data file with a name specified by the user into system memory. After the file is loaded, the program indicates the amount of free memory left.

If the DUMP function (starts at line 3000) is selected, the program asks if the data is to be dumped to the terminal or to the disk. If the data is to be dumped to the terminal, the Do-All program continues normally. If the data is to be dumped to the disk, the program jumps to a routine (starts at line 10000) that dumps the data to the disk. The name of the disk file under which the data is stored is selected by the user.

A single-dimension string array called CATEGORY$ was set up to allow storage of the transaction code as a single numeric quantity rather than a string description. When the item information is printed out, the element in the CATEGORY$ array with an element number corresponding to the transaction code is printed rather than printing the transaction code itself. Since only a one- or two-digit numeric quantity, rather than the ten- to 20-character string description, is used to indicate the transaction code, less space will be required to store the data on the disk and in memory.

Modification of this program to allow a different set of income/expense category names requires only changes to the DATA statements located at the beginning of the program. Note that the second DATA statement consists of six blank elements. This was necessary because elements 1–10 were reserved for income categories and only four categories were required in this application.

There is a problem when sorting data by date if the date information is in standard form. Consider two dates, 12/30/78 and 1/1/79. If these two dates were ranked using normal sorting techniques, the 1/1/79 would come first. This problem arises

Do-All-Plus program.

```
REM DO-ALL-PLUS PROGRAM
REM DO-ALL PROGRAM - KILOBAUD AUG,77 PAGE 84
REM SORT PROGRAM - KILOBAUD MAY,78 PAGE 100
REM T.E. DOYLE  2/4/79

  CATEGORIES=21
  MAXENTRIES=150
  ITEMS.PER.PAGE=28
  LINES.BETWEEN.PAGES=6
  DOUBLE.SPACE$="YES"
  COMMAND.LIST$="LDSPARB"

  REM INCOME CATEGORY NAMES
  DATA "DUES","HAMFEST INC.","DONATIONS","PROPERTY SALES"
  DATA " "," "," "," "," "," "

  REM EXPENSE CATEGORY NAMES
  DATA "TOWER RENT","NEWSLETTER","PHONE BILLS"
  DATA "SOCIAL ","HAMFEST EXP.","DONATIONS"
  DATA "PO BOX RENT","EQUIPMENT ","ANTENNA "
  DATA "PRINTING","MISC. DEBITS"
```

```
  REM HEADING NAMES
  DATA "TRANSACTION","AMOUNT","DESCRIPTION","DATE"

  DIM CATEGORY$(CATEGORIES),HEADING$(4)
  DIM N(3,MAXENTRIES),A$(2,MAXENTRIES)

  FOR INDEX=1 TO CATEGORIES
   READ CATEGORY$(INDEX)
  NEXT INDEX

  FOR INDEX=1 TO 4
   READ HEADING$(INDEX)
  NEXT INDEX

  PRINT "REPEATER CLUB BUSINESS PROGRAM"
  PRINT
1000 REM ENTRY POINT FOR INSTRUCTIONS
  INPUT "NEED INSTRUCTIONS";ANSWER$
  IF LEFT$(ANSWER$,1)="N" THEN 1140

  REM PRINT INSTRUCTIONS
  PRINT "THIS PROGRAM CONTAINS SEVEN FUNCTIONS"
  PRINT "TO SELECT A FUNCTION - "
  PRINT "TYPE IN THE FIRST LETTER OF THE FUNCTION NAME"
  PRINT "FOLLOWED BY A CARRIAGE RETURN."
  PRINT
  PRINT "LOAD - USED TO ENTER DATA FOR A NEW FILE FROM THE"
  PRINT "       TERMINAL OR TO LOAD A FILE FROM THE DISK"
  PRINT
  PRINT "DUMP - USED TO DUMP AN UPDATED FILE TO THE DISK"
  PRINT "       OR TO THE TERMINAL"
  PRINT
  PRINT "SORT - A FILE THAT HAS BEEN LOADED MAY BE SORTED BY"
  PRINT "       ANY OF THE ITEM CHARACTERISTICS"
  PRINT
  PRINT "PRINT - A FILE THAT HAS BEEN LOADED MAY BE PRINTED"
  PRINT "        COMPLETELY OR PARTIALLY"
  PRINT
  PRINT "ADD - ITEMS MAY BE ADDED TO A FILE THAT HAS BEEN LOADED"
  PRINT
  PRINT "REMOVE - ITEMS MAY BE REMOVED FROM A LOADED FILE"
  PRINT
  PRINT "BALANCE - TOTAL EXPENSES, TOTAL INCOME AND BALANCE"
  PRINT "          ARE CALCULATED AND PRINTED"
  PRINT
  PRINT "TRANSACTION CODE NUMBERS"
  PRINT "INCOME CATEGORIES"
  FOR INDEX =1 TO 10
   IF CATEGORY$(INDEX)<>" " THEN PRINT INDEX;"- ";CATEGORY$(INDEX)
  NEXT INDEX
  PRINT "EXPENSE CATEGORIES"
  FOR INDEX =11 TO CATEGORIES
   PRINT INDEX;"- ";CATEGORY$(INDEX)
  NEXT INDEX

1140 REM ENTRY POINT FOR NEW COMMAND
  PRINT
  INPUT "COMMAND";COMMAND$
  FOR INDEX=1 TO LEN(COMMAND.LIST$)
   IF COMMAND$=MID$(COMMAND.LIST$,INDEX,1) THEN 1210
  NEXT INDEX
  PRINT "NO FUNCTION OF THAT TYPE EXISTS IN THIS PROGRAM."
  GOTO 1000
 1210 ON INDEX GOTO 2000,3000,4000,5000,6000,7000,8000

2000 REM LOAD FUNCTION
  P=1
  INPUT "LOAD FROM DISK OR TERMINAL";ANSWER$
  IF LEFT$(ANSWER$,1)="D" THEN 11000
  PRINT
  PRINT "ENTER DATA IN FOLLOWING FORMAT"
  PRINT HEADING$(1);" , ";HEADING$(2)
  PRINT HEADING$(3)
  PRINT HEADING$(4)
  PRINT
  PRINT "TYPE 0 FOR TRANSACTION CODE AND AMOUNT"
  PRINT "TYPE $ FOR DESCRIPTION AND DATE TO STOP"
 2065 PRINT
  INPUT N(1,P),N(2,P)
  INPUT A$(1,P)
 2090 INPUT A$(2,P)
  PRINT
  IF N(1,P)<>0 OR N(2,P)<>0 THEN 2120
  IF A$(1,P)="$" AND A$(2,P)="$" THEN 2200
 2120 P=P+1
  IF LEN(A$(2,P-1))<6 THEN PRINT"DATE ERROR":P=P-1:GOTO 2090
  IF LEN(A$(2,P-1))>8 THEN PRINT "DATE ERROR":P=P-1:GOTO 2090
  IF P<=MAXENTRIES THEN 2065
  PRINT "TOO MANY ENTRIES"
  GOTO 1140

2200 REM SET UP DATE NUMERICAL ARRAY
  PRINT "FILE LOADED"
  FOR INDEX=1 TO P-1
   X$=A$(2,INDEX):GOSUB 9900
   N(3,INDEX)=X
  NEXT
  PRINT "REMAINING RAM = ";FRE
  GOTO 1140

3000 REM DUMP FUNCTION
  INPUT "DUMP TO DISK OR TERMINAL";ANSWER$
  IF LEFT$(ANSWER$,1)="D" THEN 10000
  PRINT
  GOSUB 9520
  FOR INDEX=1 TO P-1
   PRINT N(1,INDEX);",";N(2,INDEX)
   PRINT A$(1,INDEX)
   PRINT A$(2,INDEX)
  NEXT INDEX
  PRINT "0,0"
  PRINT "$"
  PRINT "$"
  GOSUB 9520
  GOTO 1140


4000 REM SORT FUNCTION
  GOSUB 9080
```

```
INPUT "TYPE # FOR SORT";T
IF T>2 THEN 4130
4050 M=P
4055 M=INT(M/2)
IF M=0 THEN 1140
J=1 : K=(P-1)-M
4070 I=J
4075 L=I+M
IF N(T,I)<=N(T,L) THEN 4105
GOSUB 9210
I=I-M
IF I<1 THEN 4105
GOTO 4075
4105 J=J+1
IF J>K THEN 4055
GOTO 4070

4130 REM SORT BASED ON DESCRIPTION OR DATE
IF T=4 THEN T=3 : GOTO 4050
T=T-2
M=P
4160 M=INT(M/2)
IF M=0 THEN 1140
J=1 : K=(P-1)-M
4190 I=J
4200 L=I+M
IF A$(T,I)<=A$(T,L) THEN 4260
GOSUB 9210
I=I-M
IF I<1 THEN 4260
GOTO 4200
4260 J=J+1
IF J>K THEN 4160
GOTO 4190

5000 REM PRINT FUNCTION
L=0
INPUT "COMPLETE OR PARTIAL PRINT ( C OR P )";COMMAND$
IF COMMAND$="P" THEN 5100
GOSUB 9800
GOSUB 9350
FOR INDEX=1 TO P-1
 GOSUB 9410
 L=L+1
 IF L=ITEMS.PER.PAGE THEN GOSUB 9600
NEXT INDEX
GOSUB 9460
GOSUB 9800
GOTO 1140

5100 REM PARTIAL PRINT BASED ON TRANSACTION CODE OR AMOUNT
GOSUB 9080
INPUT "NUMBER OF ITEM FOR LIMITS";T
IF T>2 THEN 5230
INPUT "ENTER MIN,MAX";L1,H
5160 GOSUB 9800
GOSUB 9350
FOR INDEX=1 TO P-1
 IF N(T,INDEX)<L1 OR N(T,INDEX)>H THEN 5170
 GOSUB 9410
 L=L+1
 IF L=ITEMS.PER.PAGE THEN GOSUB 9600
5170 NEXT INDEX
GOSUB 9460
GOSUB 9800
GOTO 1140

5230 REM PARTIAL PRINT BASED ON DATE
T=T-2
5240 INPUT "MINIMUM, MAXIMUM";B1$,C1$
IF C1$<B1$ THEN 5240
IF T=1 THEN 5260
X$=B1$:GOSUB 9900:L1=X
X$=C1$:GOSUB 9900:H=X
T=3
GOTO 5160

5260 REM PARTIAL PRINT BASED ON DESCRIPTION
GOSUB 9800
GOSUB 9350
FOR INDEX=1 TO P-1
 IF A$(T,INDEX)>=B1$ AND A$(T,INDEX)<=C1$ THEN GOSUB 9410
NEXT INDEX
GOSUB 9460
GOSUB 9800
GOTO 1140

6000 REM ADD FUNCTION
IF P<MAXENTRIES THEN 6040
PRINT "TOO MANY ENTRIES"
GOTO 1140
6040 PRINT "ENTER THE FOLLOWING DATA:"
GOSUB 9150
P=P+1
IF LEN(A$(1,P))<25 AND LEN(A$(2,P))<25 THEN 1140
PRINT "STRING TOO LONG-WARNING ONLY"
GOTO 1140

7000 REM REMOVE FUNCTION
PRINT "ENTER THE FOLLOWING DATA"
GOSUB 9150
FOR INDEX=1 TO P-1
 IF N(1,INDEX)<>N(1,P) OR N(2,INDEX)<>N(2,P) THEN 7160
 IF A$(1,INDEX)<>A$(1,P) OR A$(2,INDEX)<>A$(2,P) THEN 7160
 FOR K=INDEX TO P-2
  FOR T=1 TO 2
   A$(T,K)=A$(T,K+1)
   N(T,K)=N(T,K+1)
   N(3,K)=N(3,K+1)
  NEXT T
 NEXT K
 P=P-1
 GOTO 1140
7160 NEXT INDEX
PRINT "NO MATCH FOUND - NO ITEM REMOVED"
GOTO 1140

8000 REM CURRENT BALANCE FUNCTION
TOTAL.EXPENSES=0 : TOTAL.INCOME=0
```

because the year information, which is the most significant part, is in the least significant digit positions. Note that if the dates being sorted do not differ in the year positions, the list will be sorted properly.

To solve this problem, I included a routine (starts at line 9900) to convert the date information from string form to a six-digit numerical value. The most significant two-digit positions contain the year, the next two contain the month and the least significant two-digit positions contain the date. For the two dates discussed above, the numeric date values would be 783012 for 12/30/78 and 790101 for 1/1/79. This numeric equivalent date is stored on the disk file in addition to the date in string form. Since I incorporated this change, I have had no problems in sorting files by date.

Printing a list of items in the original Do-All program involved printing a heading followed by the items. If there are a large number of items, the program will print them out sequentially without leaving breaks for the start of new pages. The print function (starts at line 5000) was modified such that the program prints a heading at the top of each page and includes a break between pages.

As is, the program will print a heading and 28 double-spaced items per page. This results in 11-inch page lengths when printed on a Teletype machine (ASR-33). To eliminate the double spacing of items, change the value of DOUBLE.SPACE$ from YES, the value to which it is set at the beginning of the program, to NO. To change the number of items per page, change the value of ITEMS.PER.PAGE, which is presently set to 28 at the beginning of the program.

Form-feed action is simulated by inserting a number of blank lines between pages. On a Teletype machine six blank lines at the end of each page resulted in a page length of 11 inches. If you wish to modify the number of blank lines inserted between pages, change the value of LINES.BETWEEN.PAGES, which is set to 6 at the beginning of the program.

The Do-All program included provisions for a user-defined function. For this application the function was set up as a balance summary. This function (starts at line 8000) reads through the file information in memory and calculates total income, total expenses and net balance. The only other significant change to the Do-All program was the inclusion of several print statements within the program to make it self documenting.

This program will run under BASIC-E if you change the PRINT USING statements in the routines starting at lines 8000 and 9140 to normal PRINT statements.

Quarterly Report Program

The financial report program is designed to process a data file set up by the Do-All-Plus program. The report program prints a summary of club transactions by printing a list of income and expense category amounts by quarter. The program asks for the name of the data file containing the transaction information, the date of the report, the year the report is to cover and the starting cash balance at the beginning of the year.

After printing the income and expense information by category, the program prints total income, total expense and net balance information. With the cash balance at the beginning of the year included, the net balance total should equal the present cash balance.

Rather than reading the entire data file into memory and then processing the data, the program reads only one record from the file into memory at a time. The disk file access is a simple sequential read operation, so it was not necessary to read the entire file into memory.

After reading a record from the disk, the program determines if the date falls within the year that the report is to cover. If the date falls outside the year, the next record will be read. If the date falls within the year, the date will be converted from six-digit numerical form to a one- or two-digit month form and then to the appropriate quarter.

Store Hours:
Mon - Fri 9-5:30
Ask for spec sheets

EXTRA SPECIAL PRICES & DELIVERY

Prices good until next Magazine issue

Domestic & International
Dealer Inquiries Invited

PAPER TIGER PRINTER

Includes Graphics (New low price) . . . . $ 949
Best quality cables for TRS-80 & Apple, In stock . . . . $ 39

TI 810 SERIAL PRINTER

Shipping included on TI's $1725

TI 820 SERIAL PRINTER

Full package $1995

QDP - 100

State Of The Art **Reliable - Powerfull**
Complete - Up & Running **Powered & Programmed**

SUPERBRAIN COMPUTER

34K System $2995
64K System . . . . (CP/M & Basic included) . . . . Add $ 195

INTERTEC INTERTUBE TERMINAL

Checked and tested by us . . . . $ 797

ATARI-VIDEO COMPUTER SYSTEM 800

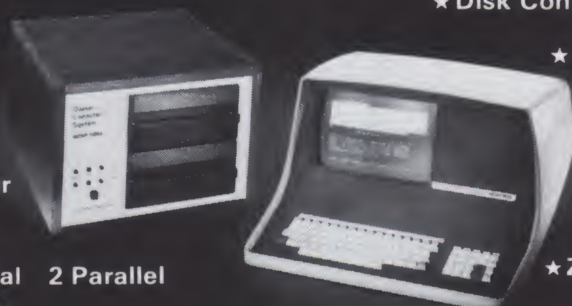
8K RAM, Basic, cassette recorder . . . . $ 895
Software & disk drives available

SINGLE BOARD COMPUTER

4MZ Z-80, DBL sided, DBL density disk controller, 2716 prom burner, 2 parallel & 2 Ser ports, real time clock, Bios for CPM 2.2, monitor in prom. This is absolutely the best board we have ever seen. It plus the Con. Data 64K memory board gives you a complete system. One year parts & labor guarantee. (The best) . . . . $ 995

QDP 100

- ★ Two 8" double-sided, double density floppies
- ★ Monitor in Prom
- ★ 64K Memory
- ★ Int. 2716 Prom Burner
- ★ Four ports
  2 Serial 2 Parallel
- ★ Full editing, smart CRT terminal
- ★ Disk Controller—up to 4 M.B.
- ★ Basic language included
- ★ S-100 Motherboard
- ★ CP/M 2.2 included
- ★ Real time clock
- ★ Z-80 4MHZ CPU



Quasar Computer System

Take a good look at the other machines on the market and compare. Our system is burned in, fully tested and assembled. The CPU Board is warranted for parts and labor for one year. Please call and we'll tell you more.
Stock to 30 days ARO $4795

CENTRAL DATA 4MZ 64K

Dynamic Ram Bd. assembled & tested
16K — **$250** 32K — **$350** 48K — **$450** 64K — **$549**

TELETEK DBL. DENSITY, DBL. SIDED

Disk Controller Board . . . . $ 395
(The best) includes cable & source bios

MFE DBL. SIDED, DBL. DENSITY

Floppy disk drives (the best) . . . . $ 650
Using the Teletek Controller under CP/M, this drive will give you almost one megabyte per disk drive
Power supply for above . . . . $ 110

SOFTWARE

Integrated Accounting System G/L, A/P, A/R, P/R, INV. $1500
Selector III Data Base Management System . . . . $ 295
Wordstar by Micro Pro Word Processing System . . . . $ 445
C Basic . . . . $ 109

QUASAR FLOPPY SYSTEM

★ Two MFE DBL sided drives ★ Cable ★ Case & Power Supply assembled and tested . . . . **Special $1695**

QUASAR 2 MEG FLOPPY

Special - $2095

- ★ 2 MFE double sided drives
- ★ Wood cabinet
- ★ Teletek disk controller board
- ★ CP/M version 2.2 & bios
- ★ Power supply & cable
- ★ Assembled & tested

WE HONOR master charge THE INTERBANK CARD

Checks, money orders accepted
Add $2.50 freight charges on orders under 10 lbs. Over 10 lbs. F.O.B. Cleveland

QUASAR DATA PRODUCTS ✓31

VISA

25151 Mitchell Dr., No.Olmsted, Ohio 44070 (216)779-9387

A two-dimension array called VALUE is used to keep a running total of the category amounts by quarter. Note that of the five pieces of information present in the record, the program only uses three—transaction code, amount and numerical form of the date. Item description and date in string form are not used.

A single-dimension string array called CATEGORY$ is used to store the names for the income and expense transaction codes. These names, which are contained in the DATA statements, should match the category names used in the Do-All-Plus program. If the category names or number of transaction codes is changed in the Do-All-Plus program, the changes must also be made in this program.

This program was written in CBASIC and could be modified to run in BASIC-E by changing the PRINT USING statements to normal print statements. Unfortunately, removing the formatted print statements would make the report printout unorganized.

The program is set up to prepare a report based on a calendar year running from January through December. The program determines if an item should be included in the report by comparing the six-digit numerical date value for the item with DATE.MIN and DATE.MAX. If the date falls between these two dates, the item is included.

The value for DATE.MIN is calculated by multiplying the value for the year as entered by the user by 10000. For example, if the user entered 78 for the year, the value of DATE.MIN would be 780000. The value for DATE.MAX is calculated by adding 1 to the year and multiplying by 10000. For the previous example, where DATE.MIN is 780000, the value of DATE.MAX would be 790000.

To change the report year to a non-calendar fiscal year, it is necessary to change the value of DATE.MIN and DATE.MAX. As an example, consider a fiscal year running from July 1 through June 30. The lines that calculate DATE.MIN and DATE.MAX would have to be changed as shown in Example 1.

```
DATE.MIN = (YEAR • 10000) + 700
DATE.MAX = ((YEAR + 1) • 10000) + 700
```

Example 1.

If the modifications were made and the user requested a report for 78, the program would set DATE.MIN to 780700 and DATE.MAX to 790700. The program would then include in the calculations any items with a date between these values. Note that items with a date greater then July 0, 1978, and a date less then July 0, 1979, will be included. The program would be modified for other fiscal years in a similar manner.

Using the Programs

The first step in using these programs is to set up a disk file containing income and expense items. First, execute the Do-All-Plus program and specify the LOAD function. When asked if data is to come from the disk or the terminal, specify the terminal. Enter the item information for all items. When you are through entering items, enter 0 for transaction code and amount and enter a $ for description and date.

Next, specify the PRINT function to obtain a printout of the data file as it exists in memory. If there are items with errors, remove them with the REMOVE function and add corrected information with the ADD function. When the data file is correct, it may be sorted by specifying the SORT function. The SORT function allows sorting by transaction code, amount, description or date. Sorting the file by date has turned out to be the most useful.

After the file has been entered, corrected and sorted, specify the DUMP function. When asked if the data should be dumped to disk or terminal, specify the disk. The program will request a file name under which the data is to be stored. Be sure that the file name you choose is not the name of a file already existing on the disk. If you wish to make a backup copy of the file on the same disk, re-

```
 FOR INDEX=1 TO P-1
  IF N(1,INDEX) > 10 THEN TOTAL.EXPENSES=TOTAL.EXPENSES+N(2,INDEX)
  IF N(1,INDEX) < 11 THEN TOTAL.INCOME=TOTAL.INCOME+N(2,INDEX)
  NEXT INDEX
  PRINT "TOTAL INCOME   = ";
  PRINT USING "$$####,.##";TOTAL.INCOME
  PRINT "TOTAL EXPENSES = ";
  PRINT USING "$$####,.##";TOTAL.EXPENSES
  PRINT "BALANCE        = ";
  PRINT USING "$$####,.##";TOTAL.INCOME - TOTAL.EXPENSES
  GOTO 1140

9080 REM PRINT HEADINGS WITH NUMBER CODE
  PRINT
  FOR INDEX=1 TO 4
   PRINT INDEX;HEADING$(INDEX)
  NEXT INDEX
  PRINT
  RETURN

9150 REM PRINTS HEADINGS AND ALLOWS INPUT
  FOR INDEX=1 TO 4
   PRINT HEADING$(INDEX)
  NEXT INDEX
  INPUT N(1,P),N(2,P),A$(1,P),A$(2,P)
  X$=A$(2,P):GOSUB 9900:N(3,P)=X
  IF LEN(X$)<6 THEN PRINT "DATE ERROR":GOTO 9150
  IF LEN(X$)>8 THEN PRINT "DATE ERROR":GOTO 9150
  RETURN

9210 REM BUBBLE SORT SWAP
  X1=N(1,L)
  X2=N(2,L)
  B1$=A$(1,L)
  B2$=A$(2,L)
  FOR Z=1 TO 2
   N(Z,L)=N(Z,I)
   A$(Z,L)=A$(Z,I)
  NEXT
  N(1,I)=X1
  N(2,I)=X2
  A$(1,I)=B1$
  A$(2,I)=B2$
  X1=N(3,L):N(3,L)=N(3,I):N(3,I)=X1
  RETURN

9350 REM PRINTS TITLES
  GOSUB 9460
  PRINT HEADING$(1);TAB(19);HEADING$(2);
  PRINT TAB(32);HEADING$(3);TAB(55);HEADING$(4)
  PRINT
  RETURN

9410 REM PRINTS ONE ENTRY
  PRINT CATEGORY$(N(1,INDEX));TAB(15);
  PRINT USING "$$####,.##";N(2,INDEX);
  PRINT TAB(28);A$(1,INDEX);TAB(54);A$(2,INDEX)
  IF DOUBLE.SPACE$="YES" THEN PRINT
  RETURN

9460 REM PRINTS A LINE OF -------
  FOR Z=1 TO 62
   PRINT "-";
  NEXT
  PRINT
  RETURN

9520 REM PRINTS A STRING OF 50 NULLS
  FOR INDEX=1 TO 50
   PRINT CHR$(0);
  NEXT INDEX
  RETURN

9600 REM NEW PAGE
  GOSUB 9800
  GOSUB 9350
  RETURN

9800 REM FORM FEED
  FOR Z=1 TO LINES.BETWEEN.PAGES
   PRINT
  NEXT Z
  L=0
  RETURN

9900 REM CONVERT DATE STRING TO NUMERICAL
  L=1
  IF MID$(X$,2,1)="/" THEN ABC$="0"+LEFT$(X$,1):GOTO 9940
  ABC$=LEFT$(X$,2):L=2
 9940 B$=MID$(X$,L+2,LEN(X$)-L-4)
  IF LEN(B$)=1 THEN B$="0"+B$
  C$=RIGHT$(X$,2)
  X=VAL(C$+ABC$+B$)
  RETURN

10000 REM SAVE DATA ON DISK
  INPUT "FILE NAME";FILE$
  FILE FILE$(64)
  FOR R=1 TO P-1
    PRINT #1,R;N(1,R),N(2,R),A$(1,R),A$(2,R),N(3,R)
  NEXT R
  CLOSE 1
  GOTO 1140

11000 REM LOAD DATA FROM DISK
  INPUT "FILE NAME";FILE.NAME$
  FILE FILE.NAME$(64)
  IF END #1 THEN 11100
  FOR R=1 TO 1000
   READ #1,R;N(1,R),N(2,R),A$(1,R),A$(2,R),N(3,R)
   P=R+1
  NEXT R
 11100 PRINT "FILE LOADED."
  PRINT "REMAINING RAM = ";FRE;" BYTES."
  GOTO 1140

  END
```

✓Reader Service—see page 257

quest the DUMP function again and dump the data file from memory to the disk under another file name.

To have a quarterly report printed, leave the Do-All-Plus program (type in a control-C). Execute the REPORT program and, when asked for the data file name, specify the file name in the DUMP function. To add items to the data file, run the Do-All-Plus program and request the LOAD function. When asked if the data is to be loaded from disk or terminal, specify the disk. When asked for the file name, specify the name used in the DUMP command.

After the file is loaded, add items to the file with the ADD function. After all new items have been added, re-sort the file if necessary and dump the updated version back to the disk using the DUMP command.

Note that the LOAD function is not to be used to add items to an existing file. If you load a file from the disk and then attempt to add items with the LOAD function, it will erase all the items in memory that were loaded from the disk. If you then dump the file back to the disk, you will erase the old items from the disk.

To avoid problems of that sort, it is best to use the PRINT function before using the DUMP function. Creating backup copies of the data file on other disks using the PIP function in CP/M is also recommended.

Another application for which these programs are well suited is personal finance. The programs have been used for this purpose by changing the income category names to:

1) Full-time job
2) Consulting
3) Equipment sales
4) Investments

The expense category names were changed to:

1) Computer equipment
2) Shelter—house payments, rent, etc.
3) Food
4) Insurance
5) Automobile—payments, insurance, gas and oil, etc.
6) Entertainment—subscription to *Microcomputing*, etc.
7) Investments
8) Savings—this category is probably not required for computer owners
9) Taxes
10) Health
11) Heat and electricity

I have found it more meaningful to keep the number of expense categories to a minimum. For example, there is one category specified for automobile expenses. This category could have been broken down into several categories such as auto insurance, auto payments, etc. By lumping all auto expense items into one category, you obtain a more representative picture of the actual expenses involved in owning a vehicle.

If you wish to save yourself the effort of typing the programs in, send me an 8 inch CP/M formatted disk with return postage and I will copy the programs onto your disk. ■

Quarterly Financial Report program.

```
REM QUARTERLY FINANCIAL REPORT PROGRAM
REM PROCESSES DATA FILE SET UP UNDER DO-ALL-PLUS PROGRAM
REM T.E. DOYLE  2/4/79

CATEGORIES=21

PRINT "QUARTERLY FINANCIAL REPORT"

INPUT "DATA FILE NAME";FILE$
FILE FILE$(64)
IF END #1 THEN 200

INPUT "REPORT DATE";REPORT.DATE$
INPUT "BALANCE AT START OF YEAR";BALANCE
INPUT "YEAR";YEAR
DATE.MIN=YEAR*10000
DATE.MAX=(YEAR+1)*10000
PRINT:PRINT:PRINT

DIM VALUE(CATEGORIES,4),CATEGORY$(CATEGORIES)

REM CALCULATE CATEGORY VALUES BY QUARTER
FOR RECORD=1 TO 1000
  READ #1,RECORD;CATEGORY,AMOUNT,W$,D$,DATE
  IF DATE< DATE.MIN OR DATE > DATE.MAX THEN 100
  MONTH=INT((DATE-DATE.MIN)/100)
  IF MONTH < 4 THEN QUARTER=1
  IF MONTH < 7 AND MONTH > 3 THEN QUARTER=2
  IF MONTH < 10 AND MONTH > 6 THEN QUARTER=3
  IF MONTH > 9 THEN QUARTER=4
  VALUE (CATEGORY,QUARTER)=VALUE(CATEGORY,QUARTER)+AMOUNT
100 NEXT RECORD
200 CLOSE 1

REM READ CATEGORY NAMES INTO ARRAY
FOR CATEGORY=1 TO CATEGORIES
  READ CATEGORY$(CATEGORY)
NEXT CATEGORY

REM INCOME CATEGORY NAMES
DATA "DUES","HAMFEST","DONATIONS","PROPERTY SALES"
DATA " "," "," "," "," "," "

REM EXPENSE CATEGORY NAMES
DATA "TOWER RENT","NEWSLETTER","PHONE BILLS"
DATA "SOCIAL ","HAMFEST","DONATIONS"
DATA "PO BOX RENT","EQUIPMENT ","ANTENNAS "
DATA "PRINTING","MISC. DEBITS"


REM PRINT HEADING FOR CHART
Y=1900+YEAR
PRINT TAB(19);"*";Y;
PRINT TAB(27);"FINANCIAL REPORT *"
PRINT
PRINT TAB(32);"QUARTER";TAB(57);"YEAR"
PRINT "* INCOME *";
PRINT TAB(22);"1";TAB(31);"2";TAB(40);"3";TAB(49);"4";TAB(56);"TO DATE"
FOR Z=1 TO 63
PRINT "-";
NEXT Z
PRINT
REM CALCULATE CATEGORY TOTALS
REM PRINT CATEGORY VALUES BY QUARTER AND TOTAL
FOR CAT=1 TO CATEGORIES
 TOTAL=0
  FOR QUARTER=1 TO 4
    TOTAL=TOTAL+VALUE(CAT,QUARTER)
  NEXT QUARTER
 IF CAT < 11 THEN INCOME=INCOME+TOTAL
 IF CAT > 10 THEN EXPENSES=EXPENSES+TOTAL
 IF TOTAL=0 THEN 400
 IF FLAG=1 OR CAT<10 THEN 300
 PRINT
 PRINT "* EXPENSES *"
 FOR Z=1 TO 63
  PRINT "-";
 NEXT Z
 PRINT
 FLAG=1
300 PRINT CATEGORY$(CAT);TAB(18);
PRINT USING "####.##";VALUE(CAT,1);TAB(27);
PRINT USING "####.##";VALUE(CAT,2);TAB(36);
PRINT USING "####.##";VALUE(CAT,3);TAB(45);
PRINT USING "####.##";VALUE(CAT,4);TAB(54);
PRINT USING "$$####,.##";TOTAL
400 NEXT CAT

REM PRINT SUMMARY INFORMATION
PRINT:PRINT
PRINT TAB(39);"START BALANCE";TAB(54);
PRINT USING "$$####,.##";BALANCE
PRINT TAB(39);"TOTAL INCOME";TAB(54);
PRINT USING "$$####,.##";INCOME
PRINT TAB(39);"TOTAL EXPENSES";TAB(54);
PRINT USING "$$####,.##";EXPENSES
PRINT
PRINT TAB(39);"NET BALANCE";TAB(54);
NET=(BALANCE+INCOME)-EXPENSES
PRINT USING "$$####,.##";NET
PRINT:PRINT
PRINT "REPORT DATE - "; REPORT.DATE$
PRINT : PRINT : PRINT : PRINT : PRINT
END
```

Financial Report sample run.

```
QUARTERLY FINANCIAL REPORT
DATA FILE NAME TEST.DAT
REPORT DATE 2/17/79
BALANCE AT START OF YEAR 100.50
YEAR 78


                   * 1978  FINANCIAL REPORT *

                               QUARTER                    YEAR
* INCOME *          1        2        3        4        TO DATE
---------------------------------------------------------------
DUES             1925.50   457.50   199.00   243.00   $2,825.00
HAMFEST           401.00  2029.61     0.00   135.00   $2,565.61
PROPERTY SALES      0.00     0.00     0.00    13.00      $13.00

* EXPENSES *
---------------------------------------------------------------
TOWER RENT        330.00   330.00   330.00   330.00   $1,320.00
NEWSLETTER        129.75    14.04   111.01    11.70     $266.50
PHONE BILLS        40.39    40.37    26.90    52.92     $160.58
SOCIAL             42.64    21.00    49.00   166.50     $279.14
HAMFEST           533.54  2384.69     0.00    89.29   $3,007.52
DONATIONS          80.00     0.00     0.00     0.00      $80.00
PO BOX RENT         0.00    20.00     0.00     0.00      $20.00
EQUIPMENT           0.00    20.17     0.00   498.82     $518.99
ANTENNAS            0.00     0.00   394.36   247.17     $641.53
PRINTING            0.00     0.00     0.00    42.51      $42.51
MISC. DEBITS        0.00     3.00     0.00     0.00       $3.00


                                       START BALANCE    $100.50
                                       TOTAL INCOME   $5,403.61
                                       TOTAL EXPENSES $6,339.77

                                       NET BALANCE      -835.66
```

✓Reader Service—see page 257

PET's Librarian

Create your own subroutine library; get automatic line numbering, too.

D. J. David
28 Rue Vicq d'Azir
75010 Paris, France

This short Librarian program suppresses two drawbacks of the present PET operating system: the inability to type instruction numbers (absence of automatic line numbering) and the lack of a subroutine library; the LOAD command does not allow you to append the loaded program to a previously entered program.

Functions

Librarian has three functions:

1. Automatic numbering of programs entered through the keyboard; you specify the first line number and the increment.

2. Partial saving of a program. The classical SAVE command allows you to only save *the complete* program as it is present in memory. Here, you can save only a part of a program (especially a subroutine) from line x to line y, as you specify.

3. Appending instructions read from the cassette to instructions already present in memory. This allows you to build a program by merging different parts or subroutines taken from different cassettes. This constitutes a complete subroutine library management system. A given subroutine may be incorporated into different programs. With function 2, any part of a program may be extracted.

When appending instructions read from the cassette, there is no constraint on line numbers. If instructions 100–200 are already present, incoming subroutine A of line numbers 300–400 will be placed after 200, while incoming subroutine B with line numbers 50–90 will be placed before 100 in memory. It is irrelevant whether you append A first and B second or B first and A second.

There are two slight constraints, but they should present no problem:

1. Any line number in the user's program must be ⩾30.

2. Only single-line instructions (~ 35 characters) can be accommodated. Several instructions (separated by :) may be present, but they must hold on a single screenline, not on a "BASIC" line of 80 characters. Anyway, this obliges the user to make clear programs. Use of abbreviated keywords may be interesting in some cases.

Instructions

Automatic numbering. Start by means of RUN 2. To the question ORG, INC ? answer giving the first line number you wish and the increment. The first line number must be ⩾ 30, and the increment must be ⩾ 1 and ⩽255. If an out-of-range parameter is given, the system will use the corresponding default value (30 for ORG and 10 for INC). For instance, the answer 0,0 return is equivalent to 30, 10 return.

Then type your instructions, without numbering, terminating each instruction by return. To exit, type RUN/STOP. If you request a LIST, you'll see that your instructions have been added after Librarian. To execute, type RUN first number and not RUN alone! You can edit your program using the standard PET method, when Librarian is not running.

Program saving. Start by means of RUN 13. To the question NAME, FROM-TO ? answer giving the name of the file you want to create and the first and last line numbers of the program excerpt you want to save. For example, PROG, 00050-01025 will save lines 50 to 1025 under the name PROG on a file that can be retrieved using the third function of Librarian. The line numbers you want to save must be given with five figures (with left zeros, if necessary), separated by a hyphen.

At the end, the saved instructions are listed on the screen. Notice that the format used when writing to tape is compatible with the third function of Librarian but is not compatible with the format used by the PET when obeying a SAVE command. So, a file written by Librarian cannot be LOADed; it can only be read by Librarian, function 3.

Program reading. Start by means of RUN 17. To the question NAME ? answer with the name of the file from which you want to append instructions to your program. This file should have been created by Librarian, function 2. As usual, you may give only the first characters of the wanted name if there is no risk of confusion.

The instructions read from the file are appended to the ones already in memory (either typed on the keyboard or read from a previous file). As soon as an instruction is read, it goes into memory at the place implied by its number, exactly as if it had been typed on the keyboard. So, there is absolutely no constraint on the order of reading. When an instruction of the same line number is already present in memory, the one which is read from tape replaces it.

This allows you to merge as many partial files as necessary to build a full program. This full program may be saved either by Librarian, function 2, or by a classical SAVE after deletion of instructions 0 to 26 (Librarian itself). At end of the read operation, the complete user program is listed.

Notice: During Librarian operation, transient printouts appear on the screen. Don't worry about them. When all is finished, a steady READY. printout will appear and the cursor will

Librarian program.

```
0 LIST00060-00080
2 INPUT"ORG,INC";DEP,INC:PRINT:PRINT:PRINT:IFINC=0ORINC>255THENINC=10
3 IFDEP<30THENDEP=30
4 AD=DEP
5 AH=INT(AD/256):AL=AD-AH*256
6 POKE1023,INC:POKE1022,AL:POKE1021,AH
7 PRINT"        ":PRINT"        ":PRINT"";:GOSUB 21:PRINT
8 POKE527,145:POKE528,145:POKE529,145:POKE530,13
9 POKE531,71:POKE532,207:POKE533,49:POKE534,50
```

```
10 POKE535,13:POKE525,9
11 PRINT" ";STR$(AD)+A$:END
12 AD=PEEK(1022)+256*PEEK(1021)+PEEK(1023):INC=PEEK(1023):GOTO5
13 INPUT"NAME,FROM-TO";A$,B$:GOSUB25
14 POKE527,71:POKE528,207:POKE529,49:POKE530,54:POKE531,13:POKE525,5
15 POKE243,122:POKE244,2:OPEN1,1,1,A$:CMD1:GOTO0
16 PRINT#1:CLOSE1:PRINT"":GOTO0 :END
17 INPUT"NAME";A$:OPEN1,1,0,A$:POKE527,71:PRINT""
18 POKE611,1:POKE528,207:POKE529,49:POKE530,57:POKE531,13:POKE525,5:END
19 POKE611,0:PRINT"":POKE527,71:IFST=0GOTO18
20 PRINT"":LIST30- END
21 A$=""
22 GETB$:IFB$=""GOTO22
23 IFASC(B$)=13THENRETURN
24 A$=A$+B$:PRINTB$;:GOTO22
25 FORI=1TO11:POKE(1029+I),ASC(MID$(B$,I,1)):NEXT
26 POKE1035,171:RETURN
READY
```

blink.

Typical session. A typical session could invoice the following operations:

Load Librarian by means of the classical LOAD command

Enter a program section by means of the auto-numbering function (RUN 2)

Append a number of subroutines taken from diverse cassettes by means of the read function (any number of RUN 17)

Store some parts of the built program on different cassettes by means of the save function (RUN 13)

Any number of type 2, 3 or 4 operations may be included in any order, thus allowing total versatility

You can, for instance, introduce instructions 100–200 from the keyboard, then read from cassette A the subroutine 30–80, then read from cassette B the instructions 300–500 as well as a new version of instruction 150, then enter from the keyboard instructions 210 to 250. You can now save on two different files instructions 100–220 and instructions 230–350. You can then execute with a RUN 310 and, at the end, save the whole program by a normal SAVE command.

My experience shows that Librarian allows significant timesavings in program preparation. ■

Hal T. Gordon
University of CA, Berkeley
Berkeley CA 94720

Instruction Sets Examined and Compared

The second part of this article starts with a look at on-chip and off-chip registers.

The "thinking power" of most CPUs is primarily in the on-chip registers, to which the logic elements have fast direct access, while memory locations (more slowly accessible via the address and data buses) serve primarily for storage. A large fraction of the instruction set (54 percent in the 8080) involves on-chip register manipulations. In one-chip LSI designs, there is a limit to the size and number of on-chip registers. The 2650 has 87 bits in its registers, the 8080 has 96, and the Z-80 "more-is-better" philosophy increased this to 208 bits.

The 6800 took the bold step of reducing the on-chip registers to 72 bits (and the 6502 carried the stripping-down to a record low of 56 bits!). It would be fascinating to know the arguments used by the 6800 designers to persuade Motorola executives to go along with this revolutionary departure from conventional design. One of them must have been that the loss of on-chip power could be compensated by adding (to the simple memory-increment and -decrement types present in the 8080 set) nine new instruction types that *directly* modify or test memory locations (CLR, COM, NEG, TST, ASR, ROR, ROL, ASL and LSR), using them as "off-chip registers." The 6502 designers backtracked, omitting the first five of the 6800 direct-memory instructions (presumably because they did not think they were valuable enough).

The big problem with doing "thinking" in external memory is that, although you now have a plethora of registers to work with, access to them is slow. If I had to guess, I would say that this is what led to the creation of *zero-page instructions* (where the op code *implies* page zero, so that only *one* address byte is needed). The 256 bytes of zero-page function as a bank of external registers. The concept is extended to its utmost by the 6502, in which 33 percent of the op codes involve zero-page.

The Z-80 designers, intent on maximizing the power of their set in every possible way, also saw the advantages of direct-memory ("off-chip register") operations. As with the 6502, they decided that not all the 6800 types were worth including. They did add all the 6800 shift and rotate instructions, plus the branch-carry rotates of the 8080 and the new RRD and RLD types, plus single-bit testing, setting and resetting instructions. Since the BCDEHL registers often serve as on-chip memory, most of these instructions were also implemented for them.

The overall gain in both on-chip and off-chip power is spectacular, although the speed (because of multibyte op codes) is not. The zero-page concept (which needs one-byte op codes) cannot be implemented, but with so many on-chip registers it is not needed. The Z-80 does its "fast thinking" on-chip.

Most CPUs have one *primary accumulator*, endowed with an exceptionally rich repertoire of operations, and a number of *secondary* ones with fewer and often different functions (sometimes, as in the stack pointer, quite specialized). The 6800 is unique in having two nearly equal primary accumulators. This means that the many op codes (56) that confer power on one are duplicated for the other.

There are also five op codes involving interaction between A and B, and three that are restricted to A, so that 61 percent of the whole set involves the accumulators. Although this uses up much of the supply of one-byte codes, it allows a kind of approximation to 16-bit operation. The existence of a 16-bit index register and 16-bit stack pointer also suggests that this idea was in the minds of the designers.

This concept was totally abandoned by the 6502, which went to the extreme of Spartan 8-bit simplicity (even its stack pointer was cut to eight bits by having stack instructions imply page one) and did everything possible to enhance the power of its one accumulator. Although its two auxiliary registers (X and Y) can do data transfers, they have almost no "thinking power" and serve largely as index registers for the many addressing modes of the accumulator. Half of the 6502 op codes involve the accumulator, mostly interactions with external memory (especially the zero-page "off-chip registers").

The 6502 is by far the most "extroverted" of all designs, since only 17 percent of its instructions command purely on-chip operations. You can recognize a 6502 machine-code listing by the high frequency of 2-byte instructions. The same operation coded for the 8080 will use mostly 1-byte instructions.

Although the Z-80 has a duplicate set of the 8080 registers on-chip, this is quite different from the 6800 concept. Only one set is "on-line" at a given moment. The last two precious new Z-80 one-byte op codes are used to *exchange* sets, one for the AF

register-pair, the other for the BCDEHL registers. Although the exchange of registers is not novel (the 8080 has three instructions of this type), the new Z-80 exchanges are an advanced feature. In effect, you can quickly insert an alternate processor into the system in a state of readiness for a different or more complicated task.

The Signetics 2650 — less well-known but no less remarkable than the other chips — is so different in its organization that it is almost in another dimension. I would guess that the basic decision was to have seven very versatile accumulators. As noted above for the 6800, conferring power heavily drains the supply of one-byte op codes. This was partly solved by organizing the six secondary accumulators into two banks of three, selected (not, as in the Z-80, exchanged) by setting or resetting one bit (RS) in a status register.

Once the concept of using status bits as auxiliary instruction bits (creating a nine-bit op code) had "broken the ice," it was probably easy to use two more status bits (WC and COM) to modify the operation of many instructions and to take the even more radical step of using the three high-order bits of the memory address to specify the addressing modes. Although the remaining 13 bits address only 8K, the address space is raised to 32K by having four selectable 8K banks. This is an anticipation of the "memory-bank-shifting" trick now coming into use to expand memory beyond the 65K limit in other systems.

Having burned so many bridges, the 2650 designers went on to an *on-chip stack* (eight 15-bit registers "pointed to" by three status-register bits) that automatically stores return addresses for subroutines and interrupts. Although Adam Osborne refers to this as "primitive," it allows extremely fast operation (a 2650 subroutine call takes only three cycles, compared to six for the 6502, nine for the 6800 and 17 for the 8080 and Z-80).

There is enough register power so that *nothing at all* is done exclusively on memory locations. Zero-page is not used as off-chip registers, but reserved for fast access to interrupt-servicing subroutines, or for short programs accessed by either branch- or jump-to-subroutine op codes that *imply* zero-page (only one address byte needed). The wealth of original, sophisticated ideas in the 2650 will profoundly influence future designs.

"The wealth of original, sophisticated ideas in the 2650 will profoundly influence future designs."

Addressing Modes

Although everyone agrees that well-designed addressing modes make programming more efficient, it seems to be hard to explain exactly *why* (and, with some of the trickier modes, even *how!*). Program memory locations are addressed consecutively by moving the program counter into the address register.

If the program byte is an op code, it is moved into the control register, and the program counter auto-increments to pick up the next program byte. Some op codes cause the next one or two program bytes to be "interpreted as data," i.e., to be moved into some *other* on-chip register(s) where (depending on what logic networks were set by the op code) they may be just stored or used in an operation. This is *immediate* addressing.

If the op code commands loading of the next two data bytes into the address register, this will provide *direct* addressing of a unique (usually non-program) memory location for the operation of that op code. All designs have this mode. It is good for occasional communication with single locations anywhere in memory, but having to specify an absolute address every time a memory operation is needed would be extremely inefficient.

In the 8080, the major addressing mode is one in which the op code *implies* that the correct memory address is stored in the HL register-pair (this may have been done by a previous load-immediate into HL), causing the content of HL to be moved into the address register. A sequence of 1-byte op codes can now access the memory location specified in HL by implying that the memory address is in HL. While the 8080 can also store 16-bit addresses in its BC and DE registers, these allow only *moves* between the primary accumulator and memory. Arithmetic and logical operations between the accumulator and memory locations are possible only with HL addressing (that's the reason for the exchange instructions between other registers and HL). The awkwardness of interaction with memory is one of the major weaknesses of the 8080, and the reason why other designs (always excepting the Z-80) have not adopted the implied mode.

In the 6800 there is a 16-bit "index register" (X) that resembles HL in that it can be loaded-immediate and incremented or decremented. However, op codes that imply X-addressing must be followed by a "displacement" byte, which is added to the "base-address" in X to generate the true address transferred to the address register. This "indexed" access to a memory location requires two program bytes, instead of only one in the 8080 mode. The gain is that any location in the address range from X to X+255 can be accessed. The difference is analogous to that between the limited moves of a pawn in chess and the freer moves of a rook.

If the displacement byte were always zero, the 6800 X would work exactly like the 8080 HL. The 6800 X also shows a close analogy with its zero-page addressing; if X were set to 0000, X-addressing would be indistinguishable from zero-page addressing. In actual use, X-indexing allows *any* 256 consecutive locations in memory to be accessed by what is, in effect, a 1-byte direct address, with the added feature that the base address can be incremented or decremented by 1-byte instructions. It is a powerful mode, and the Z-80 designers liked it so well that they added *two* 16-bit index registers (IX and IY) to their chip. Unfortunately, their use (including increment and decrement) requires 2-byte op codes, and even 3-byte ones with the bit-manipulation codes, so operation is less code-efficient and slower than the 6800 X-addressing. Even so, it is a major enhancement relative to the 8080.

Zero-page addressing exists in the 6800 and 6502 (and in the 2650, in a different form). Since a "zero-page" op code *implies* that the high-order (page) address must be set to 00, only the low-order byte need be specified in the instruction; this is a combination of implied and direct addressing. Stack addressing in all designs implies that the address exists in the stack pointer, a specialized but efficient use of the implied mode.

Both the 6502 and the 2650 use only eight-bit index registers, which contain the equivalent of the displace-

ment byte of the 6800; instructions that use them need a 16-bit base-address. The op code must be followed by two direct-address bytes (except in the 6502 zero-page-indexed mode, where only one is needed since page zero is implied), to which the index-register value is added to generate the true address.

In an indexed loop, the instruction can access consecutively (by index-register increment or decrement) no more than 256 locations because the index "wraps around." However, the index can do double duty as a loop-counter and can control the address of any number of instructions (accessing different memory areas) within the loop. Very complex operations are possible.

The 8-bit indexing eliminates 16-bit on-chip base-address registers (of which there can only be a limited number) by having base-addresses specified in the *program* locations (following each indexed instruction op code). Not only the address-range is restricted (to 256 consecutive locations), but so is the base-address; although you can write a RAM program that will modify its own addresses, this is a dangerous game, and not possible for a program in ROM.

This limitation of direct-indexed addressing is overcome in the 6502 and 2650 by *indirect*-indexed addressing (absent in the 8080, Z-80 and 6800, although these sets can emulate it). The true base-address of an indirect-indexed op code is stored in two contiguous RAM locations that function as a 16-bit "off-chip register." The op code must be followed by two direct-address bytes (cut to one in the 6502 because page zero is *always* implied) that specify the location of the "off-chip address register" in RAM. The operation picks up this "indirect" address (first the low, then the high) and adds the index value (in the on-chip index register) to generate the true memory address.

Although fully automatic, all this work takes time, so execution is slower (by two cycles in the 2650, but only one in the 6502) than direct indexing. But the addressing capability is now limitless, since the "indirect address register" is modifiable at will. In fact, this is a device that allows you to create in RAM as many 16-bit base-address-remembering registers as you may require, and is conceptually similar to the 6800/6502 transformation of zero-page locations into 8-bit "off-chip registers." Although access to them is slow, the supply is far greater than that available on-chip, even in the Z-80.

This may be why Adam Osborne believes that 8-bit indexing is "more powerful" than the 6800/Z-80 16-bit indexing mode. However, it seems to me that there are complex trade-offs such that 16-bit indexing will be faster and more efficient in some operations.

In every design, the program counter is a potential 16-bit base-address register, and all (except the 8080) use it as such in relative-branching. Only the 2650 extends the concept of PC-relative addressing to other instruction types, which can access memory locations *in the program area* by a one-byte address (added to the program counter).

There is some operational resemblance to the zero-page concept, but programs carry their "personalized off-chip registers" along with them wherever they may be located in memory. This extraordinary mode needs careful structuring; most of its power is lost if the program is "frozen" in ROM, and this may be an instance where the 2650 designers' imagination simply ran wild!

The Status Register

It is possible for a "thinking" instruction to be encoded so that it will cause a program skip, branch or jump if its operation yields a special state (such as all the bits in a register becoming zero). The DJNZ (Decrement B and jump relative if B≠0) of the Z-80 is of this type. However, if every possible "thought" were to be co-encoded with every possible "action," the instruction set would become very complex. It is more practical to cause each "thinking operation" to set or reset one or more status flag bits to control the operation of *subsequent* jump instructions.

Although these bits are usually independent, it is convenient to have them in a *status register*. One reason is that whenever a running program is interrupted, its current status must be saved (usually in the stack) because the interrupt-servicing program may alter some status flags; before the interrupted program is reentered, the prior status must be *restored* for it to work properly. Also, it is often useful to check the content of the status register when you're tracking down bugs in a program.

There are variations in the status flags used by different designs. The common ones are the all-bits *zero* flag, the *sign* flag (set if bit 7 of a register is set to 1 by an operation) and the *carry* flag (a kind of "ninth" bit, set or reset by a variety of arithmetic or shift or rotate instructions). The 8080 also has a *parity* flag (set if the number of 1 bits following an operation on a register is even, and otherwise reset); the Z-80 retains this flag but assigns to it two quite different meanings (*parity* in non-arithmetic operations, but *overflow* in arithmetic ones) that in practice never conflict.

The 6800, 6502 and 2650 have no parity flag, but all have the *overflow* flag (set if there is a carry out of bit 6 in "signed binary arithmetic," where only bits 0 to 6 are numeric, bit 7 being the sign flag, + if 0 and − if 1). In all but the 2650 (as usual, somewhat eccentric) these flags (zero, sign, carry, parity and/or overflow) are individually "testable" by a pair of instructions (jump-if-flag-set and jump-if-flag-reset).

There are also "non-testable" flags that control the operation of non-jump instructions. In the 8080 only the *auxiliary carry* (out of bit 3 in arithmetic operations), which is useful for decimal arithmetic, is in the status register. Other flags (such as the *interrupt-inhibit*, which disables the interrupt pin of the processor) remain invisible on-chip. Unlike the testable flags, which are automatically set or reset by "thinking" operations, the non-testable ones are set or reset only by special instructions.

However, there are also special instructions to set or reset some of the testable flags. Thus you find (in all but the 2650, which has unusual instructions for program control of any or all flags) a specific set-the-carry instruction, highlighting the major role played by the carry in "thinking" operations. The 6800 and 6502 also have a clear-the-carry, but the 8080/Z-80 have a complement-the-carry (set it if clear and clear it if set).

The 8080 CMP A instruction (very little used, since it

"There are also 'non-testable' flags that control the operation of non-jump instructions."

Subscription Problem?

Kilobaud Microcomputing does not keep subscription records on the premises, therefore calling us only adds time and doesn't solve the problem.

Please send a description of the problem and your most recent address label to:

Kilobaud Microcomputing
Subscription Dept.
PO Box 997
Farmingdale, NY 11737

Thank you and enjoy your subscription.

compares the accumulator to itself!) *could* be used to clear the carry (but also set the zero) flag; this is one of many possible examples of how "missing" instructions can be emulated.

In most designs, the testable flags are not affected by move instructions. In the 6800, most moves affect the zero, sign and overflow flags. The advantage is that a program "learns" something about a bit-pattern whenever it handles it. The disadvantage is that these flags become highly "volatile," so that a status from a previous "thinking" operation is lost even though it may be needed later. The 6502 design compromised: Only the zero and sign flags are affected, and only by *moves into* the on-chip registers.

Nevertheless, status-saving is very important. The 6502 has instructions that push or pull the status register directly into or out of the stack. The 6800 status-save is more awkward, since the status register content must go through the accumulator to get into or out of the stack. The 2650 status-save is similar (but since it lacks a conventional stack, the save is in some other memory location). The 8080/Z-80 cannot save *only* the status register; their PUSH and POP instructions must simultaneously (and slowly) save and restore the accumulator.

The many "thinking" instructions are, in effect, "set-flags-if" instructions. Much of the art of programming consists of using these in many different ways (ranging from the obvious to the fiendishly clever) so that each of an infinite variety of possible conditions gets translated into some unique status of from one to four testable flags (with no more than 16 distinct conditions able to exist at a given moment). For truly complex operations, this is very restrictive; this limitation is overcome by constructing in memory specific *decision tables* for each problem, using as many bits as may be necessary, and creating switching networks of far greater complexity than those within the CPU, and acting like "off-chip status registers."

One of the more promising innovations is the use of status bits to alter the interpretation of an op code. This is exemplified by the "decimal flag bit" in the 6502 status register. When this bit is set, all 16 add-and-subtract op codes do automatic decimal-adjusting. In effect, this bit doubles the number of arithmetic op codes. Other designs need a decimal-adjust instruction after each add or subtract instruction so that arithmetic loops run more slowly.

The Signetics 2650 uses its WC status bit to cause its add, subtract and rotate instructions to work either with the carry (WC set) or without it (WC clear). This kind of enrichment of an instruction set is likely to be more widely adopted than the Z-80 multibyte op codes. Its implementation requires a lot of thought, since programsetting of status bits becomes a nuisance if it has to be done frequently!

Things to Come

The 8080 and its rivals were made possible by LSI technology, although the Z-80 is an early product of the new era of VLSI (very large-scale integration). VLSI is now primarily being used for designs that combine a CPU, I/O, ROM and some RAM all on a single chip (Intel 8048, Mostek 3870, etc.). These are special-purpose controllers aimed at a mass market of millions of units.

Those who have read the articles on Microelectronics in *Scientific American* (September 1977) know what a fantastic *technical* achievement a VLSI chip represents. The *intellectual* achievement of creating a superior VLSI CPU chip will be far more formidable. The Z-80 strikes me as an awkward first try, a necessary first stage of a new learning curve. The 8080 (following an amazingly fast learning curve up from the 4004) was a work of creative genius in LSI. By comparison, the Z-80 is only an immense add-on, achieving much greater power but losing in elegance. This is hard to define: Many elements, all essential, all precisely right, fit into one perfect entity. The Taj Mahal is elegant; the Pentagon is not.

> "Most of the creative ferment of the past decade was fostered by *small* companies."

Everyone knows that no existing design even comes close to being the ultimate one, and that far superior ones will become available in the near future. Before trying to guess (I have no insider knowledge!) what they will be like, we may wonder whether it is possible for giant corporations to harbor and nurture the highly creative and individualistic minds to whom we owe the revolutionary microprocessor designs. Most of the creative ferment of the past decade was fostered by *small* companies. Intel has become a giant, Zilog is backed by giants, MOS Technology was taken over by Commodore, Signetics by Philips Eindhoven.

Although large organizations can provide large resources, they are directed by an entirely different kind of mentality. It can be argued that giantism crushes creativity (in the automotive field, where do we find front-wheel drive with CVCC and rotary engines?) and replaces genuine value with shiny packaging and advertising.

Another crucial question is: Has VLSI made the 16-bit CPU the wave of the future? Here we have already seen the major minicomputer manufacturers (DEC and Data General) trying to counter the threat of major IC manufacturers (especially Texas Instruments) by downshifting to single-chip CPUs, such as the LSI-11 and microNova, in the hope that their highly developed operating systems and other software will give them a decisive competitive advantage.

Anyhow, it seems to me that Intel will probably not do any major redesign of the 8080 (beyond the enhancement at the electronics level in the 8085). The effective power will be enhanced by new LSI support chips (i.e., the 8275 CRT-controller and 8279 KB-I/O interface) that relieve the CPU of much drudgery, freeing it for thinking. The familiar stacks of PC boards, loaded with armies of chips, are on the verge of obsolescence. If Intel should decide to compete against the Commodore PET, etc., its entry could be based on a mere *handful* of chips, with a price/performance ratio that would be hard to beat.

As I stated at the start of this article, there are not many tasks that the 8080 instruction set cannot do very well, so resting on these laurels may well be the right strategy (and the Intel executives have so far proved to be skillful strategists). Although I have not carefully studied the 8048 design, which has many resemblances to the 8080, it is likely to be a strong contender at the lower level. This leaves the new 16-bit 8086, which I expect to be as revolutionary as the 8080 was, simply because it is sure to be very far up on the

✓Reader Service—see page 257

learning curve of microprocessor design.

In anticipation of the 8086, it is a certainty that Intel competitors are engaged in intensive design efforts, which may not be crystallized until the full power of the 8086 is known. The Z-80 designers (who were the 8080 designers), at last free of their desire to retain compatibility with the 8080, are sure to try to create something in VLSI that will not only be big but also wonderful. The dilemma is: Should you go whole-hog for 16-bit operation (minicomputer-style) or allow variable-bit-operation (as the TMS 9900 does)? Many operations don't really *need* 16 bits.

By now, most people know that the 6502 design arose from the discontent of some 6800 designers. They retained some 6800 elements, dropped others and added new ones, but left 105 op codes unused (consciously labeling them for "future expansion"). That this set — in an obviously unfinished state — can compete effectively with more complex ones proves the overriding importance of architecture and speed over mere size.

It is hard to guess what direction the VLSI expansion of the 6502 will take, since so many options are wide open. Nevertheless, the "feel" of the 6502 suggests that it will not add as many new on-chip registers as the Z-80. Its extreme one-accumulator orientation suggests that this will be expanded to allow many 16-bit operations. If the original designers are still involved, its new instructions are likely to add quite generalized power, not like the specific subroutine type of the Z-80 block move or search.

Motorola has already done some enhancing of the 6800, though neither the 6801 nor the 6802 is a major upgrading of the kind that competitive pressures will eventually require. If I were a Motorola executive (luckily for them I'm not!), I would be wondering whether the twin-accumulator concept, which would have to be retained in a compatible upgrading, is viable. As for the 2650, that I (and surely many others) admire its ingenuity cuts very little ice since it has not sold well, and Philips executives must be pondering its fate.

In the next few years, imponderables such as the economic climate and the intuition of executives will play a more important role than the brilliance of designers (in whose honor I wrote this article). Peering somewhat farther into the future, I think it likely that the creative genius of the Japanese, who have been making giant strides in computer technology, will be entering the picture.

Existing designs have enough momentum so that they will persist for quite a while. Some differences between them were revealed in the BASIC timing comparisons by Rugg and Feldman (June and October 1977 *Kilobaud*). Such tests may reveal more of the moronic nature of BASIC than of the ultimate power of a microprocessor. The efficiency of even the finest instruction set gets degraded by older, human-oriented high-level languages like BASIC or FORTRAN. Only a stupendously large and costly optimizing compiler program can translate them into efficient machine language.

In a talk at the 1977 WESCON, Carol Anne Ogdin referred to these older languages as "dinosaurs" and stressed the superiority of newer ones such as PASCAL and FORTH. The microcomputer revolution will not be completed until the gap between computer language and human language is bridged, even though the gap between personal computers and megacomputers is narrowing rapidly.

Professional programmers (a very special breed) now function as the language bridge, and have not yet succeeded (though IBM tried) in writing computer programs to replace themselves! Unless human language evolves into something more logical and less ambiguous, perhaps they never will; but it is the next great challenge to creativity. What's the good of having a genie at your beck and call, if you can't tell it what to do? ■

An early view of the next generation. The above was completed before any of the new designs had materialized. Several are now available, but it is not possible here to do any in-depth evaluation. With *thousands* of possible op codes, the 8086 looks impossibly complex to me. It will encounter competition from 16-bit rivals such as the Zilog Z-8000 and Motorola M-68000, and other manufacturers may risk entry into a small and overcrowded market. All these hardware marvels lack the tested software that can make them useful.

One hint that the day of the 8-bit machine is not over is the forthcoming Intel 8088, a dual 8-bit processor on one chip that will include much of the advanced thinking of the 8086. Intel executives apparently decided they needed something better than the 8080/8085 to stay competitive in the low-cost microcomputer market.

A formidable competitor is the Motorola 6809, with enhancements so great that I feel its performance will, in most areas, excel that of all older designs. It retains the twin accumulators of the 6800, but some instructions use A and B as if they were one 16-bit accumulator. It adds a second index register (Y, exactly like the 6800 X) and a second stack pointer (U, similar to the 6800 S but never used for automatic storage of return addresses or registers). Both stack pointers can also serve as index registers.

This is part of an enrichment of addressing modes. However, as in the Z-80 the modes must be specified by a second op-code byte, sacrificing speed for power. Probably to compensate for this loss, the fast "zero-page" addressing mode has been made more flexible: the "page" address is stored in a new 8-bit programmable "direct-page register," so that *any* page in memory can be addressed as if it were the 6800 zero-page.

Also, direct-page addressing, needing only one op-code byte and one address byte, has been made indexable as in the 6502. PC-relative addressing, as in the 2650, is implemented, together with "long" (16-bit) PC-relative branching, so that programs can be written to run *anywhere* in memory. Many of the valuable ideas of 6800 competitors have been adopted and extended.

I have not seen the full 6516 instruction set, which is said to use only one-byte op codes. This will often mean higher speed, but 105 new op codes cannot yield as much capability as the much greater increase in the 6809. In the coming generation, we may see several chips with very different instruction sets, none so superior that it can crush all others. The older designs, however, have become obsolete.

Lloyd A. Case
Wayne County Community College
5435 W. Fort
Detroit MI 48209

Robert M. Jennings
Indiana University Southeast
4201 Grant Line Rd.
New Albany IN 47150

Partnership Liquidation—Before the Fact

The end of a business relationship need not be painful.

Many accounting problems can be placed in computer-program format for solution purposes. These programs can range from simple to complex. To illustrate several such problems and their solutions, this article briefly lists the idea of depreciation correction (simple), and then goes on to discuss partnership liquidation scheduling (moderately complex) in some detail.

The need for depreciation correction would be a frequent one. There is an income tax advantage in accelerating the depreciation of fixed assets. The Internal Revenue Service supervises such depreciation by recommending guidelines for the expected life cycle of almost all types of depreciable assets. A firm may depart from such guidelines, but must be prepared to justify that departure. Logically, the firm would desire a shorter life option but shun a longer life option. The departure towards a shorter life cycle could be justified, for example, by multiple shift operations; extreme physical conditions of heat, cold, dust; frequent operator changes.

The need for the partnership liquidation program would be infrequent. At the same time, it would be desirable from the viewpoints of the partners to have such a schedule prepared at regular intervals. This would permit a frequent review of the relative standing of each partner.

The liquidation of a partnership form of business has been found for many years to be a practical problem and an academic exercise. There are two ways of looking at the liquidation: the scheduling of cash distribution as cash becomes available from the piecemeal sale of assets and the prior preparation of a cash schedule indicating how cash will be distributed in the event of liquidation. Liquidation of an existing business entity is usually a time of difficulty. It is possible that tempers are shorter and patience and tolerance, in general, are beginning to run at a low ebb. It would seem that the second method may be preferable from the viewpoint of human relations.

There are two divergent forces pulling at the participants in such a liquidation. One force is the strong desire to hold off on the piecemeal sale of any asset until all parties are satisfied that they have managed to get the best price under the liquidation circumstances. The other force is the strong psychological one hinted at in the previous paragraph, i.e., the desire to terminate the painful process quickly.

In essence, the first method takes a look at the current cash balance and then allocates it to the partners after some computations using the current capital account balances, the profit/loss sharing ratios are the remaining value of noncash assets. It presumes that all liabilities have already been paid (or are arranged to be paid first). The noncash assets are assumed to represent the maximum amount of any future potential loss.

The influence of the future potential loss in terms of a cash distribution schedule is immense. It must be incorporated into any computation relating to the distribution, just as any real loss (or gain) during the sell-off process must be included.

The CPA exam occasionally used an involved serial liquidation problem with a resulting vicious circle type of computation. Fortunately, the problems of this type are of the iterative type, in which a similar set of steps is followed for each cash sale of assets and cash distribution to partners. The iterative nature of the problem makes it an excellent candidate for computer programming.

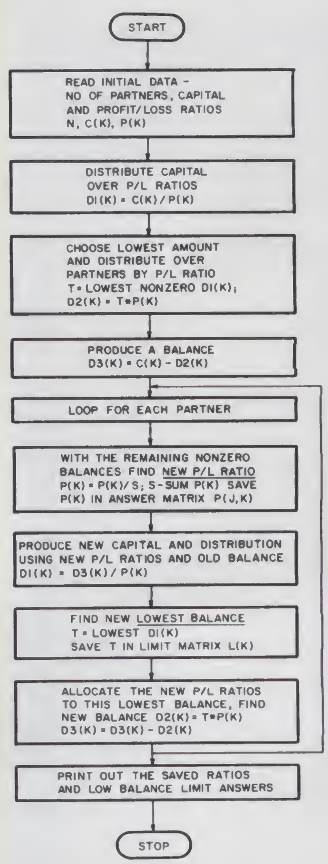


Fig. 1.

The purpose of this brief article and accompanying illustrative material is to demonstrate the second method of preparing the schedule of cash distributions. This second method has several possible advantages. From the human relations viewpoint, it should smooth the problems that could arise by informing every partner, before the first sale of any asset, exactly when and to what dollar extent he or she would participate. Another possible advantage is the objective method of determination, which can be traced and examined before the first installment is paid.

Both methods are inherently similar. The second method starts from a slightly different premise. It recognizes that there are various levels of potential future losses that would wipe out various partners. Clearly the partner who would be eliminated by the smallest of the potential future losses is in the worst situation vis-à-vis his fellow partners.

As an arithmetic example, a $100 loss would wipe out a 40 percent profit/loss sharing partner who had a capital balance of only $40. One of his partners who had a capital balance of $25 but who had only a 20 percent profit/loss sharing ratio would not be eliminated by the loss of $100 (loss is presumed as actual

```
LIST
10   READ N
20   FOR K=1 TO N
30   READ C(K)
40   NEXT K
50   FOR K=1 TO N
60   READ P(K)
70   D(K)=P(K)
80   D3(K)=C(K)
90   D1(K)=C(K)/P(K)
100  NEXT K
110  T=999999
120  FOR J=1 TO N
130  FOR K=1 TO N
140  IF D1(K) > 0 THEN 160
150  GO TO 200
160  IF D1(K) < T THEN 180
170  GO TO 200
180  T=D1(K)
190  M=K
200  NEXT K
210  L(J)=T
220  FOR K=1 TO N
230  D2(K)=T*P(K)
240  D3(K)=D3(K)-D2(K)
250  NEXT K
260  P(M)=0
270  FOR K=1 TO N
280  S=S+P(K)
290  NEXT K
300  FOR K=1 TO N
310  IF S=0 THEN 420
320  P(K)=P(K)/S
330  A(J,K)=P(K)
340  IF P(K)<> 0 THEN 370
350  D1(K)=0
360  GO TO 380
370  D1(K)=D3(K)/P(K)
380  NEXT K
390  S=0
400  T=999999
410  NEXT J
420  PRINT"      PARTNERSHIP LIQUIDATION BEFORE THE FACT"
430  PRINT"      $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
440  PRINT
450  PRINT
460  PRINT"INVESTOR","CAPITAL","PROFIT/LOSS RATIO"
470  FOR K=1 TO N
480  PRINT K,"$"C(K),D(K)*100"%"
490  NEXT K
500  PRINT
510  PRINT
520  PRINT"      LIQUIDATION SCHEDULE"
530  PRINT
540  I=1
550  FOR K=N TO 1 STEP-1
560  FOR J=1 TO 10
570  S1=S1+A(J,K)
580  NEXT J
590  IF S1=0 THEN 670
600  PRINT" ROUND "I" LIMIT: $"L(K+1)
610  FOR J=1 TO N
620  PRINT"  INVESTOR "J" TAKES "A(K,J)*100"%"
630  NEXT J
640  PRINT
650  S1=0
660  I=I+1
670  NEXT K
680  PRINT" ROUND "N" LIMIT: ANY REMAINING"
690  FOR M=1 TO N
700  PRINT"  INVESTOR "M" TAKES "D(M)*100"%"
710  NEXT M
720  DATA 4
730  DATA 70,120,80,30
740  DATA .15,.3,.35,.2
750  STOP
760  END
```

Note: This program was written in Control Data BASIC Version 2.1 for a CDC 6600/ CYBER 172 system. The 2.1 reference manual informs us that the BASIC 2.1 compiler is a modification of the Control Data BASIC Version 2.0 compiler for CYBER 70, models 72, 73 and 74; CYBER 170, models 172, 173, 174 and 175; and 6000 series computers. This version is an extension of the original Dartmouth BASIC.

```
        PARTNERSHIP LIQUIDATION BEFORE THE FACT
        $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$


INVESTOR        CAPITAL         PROFIT/LOSS RATIO
1               $ 70            15%
2               $ 120           30%
3               $ 80            35%
4               $ 30            20%

      LIQUIDATION SCHEDULE

 ROUND  1 LIMIT: $ 10
  INVESTOR  1 TAKES  100%
  INVESTOR  2 TAKES  0%
  INVESTOR  3 TAKES  0%
  INVESTOR  4 TAKES  0%

 ROUND  2 LIMIT: $ 77.14286
  INVESTOR  1 TAKES  33.33333%
  INVESTOR  2 TAKES  66.66667%
  INVESTOR  3 TAKES  0%
  INVESTOR  4 TAKES  0%

 ROUND  3 LIMIT: $ 62.85714
  INVESTOR  1 TAKES  18.75%
  INVESTOR  2 TAKES  37.5%
  INVESTOR  3 TAKES  43.75%
  INVESTOR  4 TAKES  0%

 ROUND  4 LIMIT: ANY REMAINING
  INVESTOR  1 TAKES  15%
  INVESTOR  2 TAKES  30%
  INVESTOR  3 TAKES  35%
  INVESTOR  4 TAKES  20%
+Stop!
+ At Line "750" in Program "DRCASE"
+ Program Ends
:
```

Program listing and sample run.

or potential). For this latter partner it would require an aggregate loss of \$125 (\$25 ÷ 20 percent). The second partner is clearly in a superior position in terms of the relative ability to absorb losses.

Fig. 1 and the program listing and sample run utilize the partners' capital balances at the present moment and the partners' profit/loss sharing ratios as their initial input. Liabilities are assumed as already liquidated. Any unusual gains that may arise in the liquidation process are automatically provided for.■

✓Reader Service—see page 257

Jerry Dunmire
575 S. Rengstorff Ave., #4
Mt. View CA 94040

Indexing for the PET

Did you know there's a solution to the lack of an index counter for the PET tape recorder?

After using Commodore's Personal Electronic Transactor (PET) for even a short time, you realize that the PET has no index counter for the tape recorder. This does not appear to be much of a problem until you spend 20 minutes looking for the third program on a 45-minute audio tape.

There is a solution to the missing-index-counter problem. The solution will save you time and may even save you money. By using software and the tape-transport-control capabilities built into the PET, it is possible to efficiently store several programs on each side of an audio cassette.

When Program A is stored at the beginning of a tape, all the other programs on the tape can be accessed quite easily. This seeming bit of magic is accomplished by using the PET 2001's internal clock and tape recorder controls.

The index program (Program A) provides all of the instructions for the operator. These instructions are along the same format as the PET operating system (e.g., PLEASE PRESS PLAY ON TAPE #1). A detailed explanation of the index program follows. If you want to see a sample of what the program does, enter the index program and the DATA statements in Program B.

The Program

The first few lines of the index program are devoted to displaying the contents of the tape. Each program on the tape is named, and this name is placed in a DATA statement. When the program begins execution, it fetches the number of programs stored on the tape from the DATA statement. This number is used to limit the loop in lines 110-130. The loop reads the name of each program from the DATA statement and prints it on the screen with a reference number. The reference number is used to find the index for the selected program.

A sample of the DATA statement is shown in the sample run. The extra index number in the statement is a reference to the next available area of tape.

Lines 210 and 235 in Program A are concerned with controlling the tape recorder. You can see that there are three memory locations used in controlling the tape recorder. Location 59408 is used to detect when a tape function button (e.g., F.FWD, REW, PLAY) has been depressed. If the number in location 59408 is 233, then a function button has been pushed.

When 69 is POKEd to memory location 59411, the tape machine is turned on. The recorder can be turned off by POKEing 61 to location 59411. However, there is a catch: unless location 519 contains the value 52, location 59411 has no effect on the recorder. Still another qualification is that a tape function button must be pressed before location 519 can be POKEd with 52.

Even with the tape control and the list of programs, an index number for each program is needed. The index program listed here uses time rather than a mechanical measurement to find each program. PET BASIC makes it easy to measure time with the TI function, which returns the number of jiffies (1/60 of a second) that the computer has been on. The program indexes each file by the number of jiffies it takes to

```
100 READ N:PRINT"c
110 FOR I=1TON
120 READ F$:PRINTI,F$
130 NEXTI
135 PRINT"ENTER THE NUMBER OF YOUR SELECTION
140 INPUT S
150 FOR I=1TOS
160 READ A:NEXTI
165 GOSUB500
170 PRINT"cREWIND THE TAPE AND THEN PRESS ANY KEY
180 GETA$:IFA$=""GOTO180
190 GOSUB500
200 PRINT"qqPRESS FAST FORWARD ON TAPE #1
210 IF PEEK(59408)<>233 GOTO 210
220 S1=TI
230 IF A<>ABS(TI-S1) GOTO230
235 POKE519,52:POKE59411,61
240 RESTORE
250 FOR I=1 TO S+1:READF$:NEXTI
255 GOSUB 500
260 PRINT"cTHE TAPE IS NOW AT THE START OF
265 PRINTF$
270 PRINT"PRESS SHIFT&RUN/STOP OR TYPE LOAD
280 END
500 REM THIS SUBROUTINE CHECKS TO SEE
510 REM IF THE STOP BUTTON ON TAPE #1
520 REM HAS BEEN PUSHED, IF NOT,THEN
530 REM IT PROMPTS THE OPERATOR.
531 REM
540 IF PEEK(59408)=233 GOTO 560
550 RETURN
560 PRINT"cPLEASE PRESS STOP ON TAPE#1"
570 IF PEEK(59408)=233 GOTO 570
580 RETURN
READY.
```

Program A.

q	CURSOR DOWN
r	REVERSE FIELD ON
s	HOME CURSOR
}	CURSOR RIGHT
a	CURSOR UP
b	REVERSE FIELD OFF
c	CLEAR/HOME
m	CURSOR LEFT

Table 1. Special characters.

```
300 DATA o, BREAKOUT,"STAR WAR TRAINER"
310 DATA BLACKJACK,"MAXIT 1.2","STAR TREK o.2"
320 DATA "LUNAR LANDER",500,1300,2100,2900,3700,4500,5300
```

Sample run.

reach the file using the F.FWD speed. The beginning of the tape is used as the reference for all measurements.

A good question at this time would be, "How many jiffies should I allow for each program on the tape?" Program B should help to answer this question. Tape Measure (TM) should be stored on tape before it is executed. To calculate the amount of memory that it uses, TM requires you to enter the amount of memory you have available on your system for program storage. The amount of room required to store TM on tape is measured at the F.FWD speed, and this value is used to calculate the index numbers for a 1K file and for the full memory.

With the information from TM, the index program can be used in two ways. The audio tape can be divided into equal sections that will hold a program requiring all of your memory. The other alternative is to allow just enough room for each program.

```
10 GOSUB500
15 B=FRE(0)
17 INPUT"cENTER THE AMOUNT OF MEMORY YOU HAVE FOR PROGRAM STORAGE";B1
18 B=B1-B
20 PRINT"cTURN YOUR TAPE OVER,BUT rDO NOTb REWIND IT
30 PRINT"qqPRESS F.FWD,AND WHEN THE TAPE REACHES   THE END PRESS ANY KEY
40 IF PEEK(59408)<>233 GOTO40
50 T=TI
60 GETA$:IFA$=""GOTO60
70 T=TI-T
72 POKE519,52:POKE59411,61
73 GOSUB500
74 A=INT(T*1024/B)+1:C=INT(A*B1/1024)+1
80 PRINT"cTHIS PROG OF";B;"BYTES TOOK UP";T
90 PRINT"JIFFIES OF TAPE-TIME.
100 PRINT"THIS CORRESPONDS TO";A;"JIFFIES":PRINT"PER 1K BYTES."
110 PRINT"qTO STORE A PROGRAM REQUIRING ALL OF
120 PRINT"YOUR MEMORY WILL REQUIRE";C;"JIFFIES.
200 END
500 IFPEEK(59408)=233 GOTO560
510 RETURN
560 PRINT"cPLEASE PRESS STOP ON TAPE#1"
570 IF PEEK(59408)=233GOTO570
580 RETURN
READY.
```

Program B.

When the tape is divided into sections that will hold your largest program, there is always room to modify and restore the program—without moving or destroying other files. Allowing just enough room to store a given program might allow for more programs per tape, so each method has some advantages.

If maximum storage density is a concern, Tape Measure does not give an accurate result. This is because TM is a short program, and when it is stored on tape the leader takes a significant amount of tape in relation to the program. This error can be removed either by measuring the leader or using a longer program to estimate your storage requirements. Care should be exercised so that you don't get your files too close together or you will end up with lost programs.

I hope that these programs will be as useful to you as they have been to me. ■

✓Reader Service—see page 257

Are You Bugged by Memory Errors?

This program for 8080/Z-80 systems will test memory and isolate both hard and soft errors.

Option	Meaning
1	Loads the portion of memory under test with 0s, waits briefly and then checks each location for 0s. It generates a display of each failed location and builds a memory map of the error locations.
2	Same as option 1 except that 1s are used in place of 0s.
3	Same as option 1 except that a test byte is loaded and checked. The test byte can be any hex byte and is inserted by the operator.
4	Rechecks the last test run. It does not reload the memory; it only checks to see that the last load is still valid. It displays error locations and builds a memory map.
5	Performs a continuous check of each location under test and displays error locations. The memory map function is not used in this test.
6	Prints the memory map generated by tests 1,2,3 and 4.
8	Allows the memory test area to be redefined.
9	Returns the program control to the system monitor.

Table 1. Input option selection.

```
G CCCC
INPUT MEMORY TOP HEX BYTE>>9>F
INPUT MEMORY TEST START LOW BYTE>>0>C
INPUT MEMORY TEST START HIGH BYTE>>9>0
```

Fig. 1. Memory boundary set display.

John R. Stanton
PO Box 14393
Austin TX 78761

This assembly-language memory diagnostic program will allow you to put your memory chips and boards through their paces anytime you suspect a fault. It is meant to reside in 1K of EPROM and needs only 256 bytes of RAM in order to function. Assembled using Michael Shrayer's ESP-1 assembler, it can be adapted, with appropriate changes, for assembly on most of the assemblers available for the 8080 and run entirely in RAM if desired. ESP-1 is marketed for a number of systems including the TRS-80.

The program itself is written using subroutines and can be modified to meet the special needs of most systems. When running, it allows the boundaries of the memory under test to be defined, prints out the bad locations and builds a graphic map of the bad locations in your memory. Unlike many other diagnostic programs, it provides a variety of testing options and features that make it easy to use and allows quick identification of the trouble location.

Memory Failure Modes

Two types of failure are common to the RAM memory chips in use today: those that cause the hard loss of ability to change the state of a bit or bits and those soft failures that allow a change but revert back to the original state after a period of time. The soft failures are the most difficult to detect because they require time to pass before they can be detected. No single program can detect all types of the hard and soft failures because a different philosophy must be employed for each.

The programs used to detect hard failures usually write a bit pattern into each memory word and then check each word to see if it contains what was written. The sophisticated versions of this type of test will load a different bit pattern in each word on the first pass, check each word on the second pass and continue to load and check until every possible bit pattern has been written into every word.

This test will catch most of the hard failures and some of the soft failures. This type of test may not detect those soft failures that occur after awhile or those failures that are overwritten by the test program.

Soft failures in dynamic memory chips are usually associated with a refresh problem—either the inability of a single cell to be refreshed or a deficiency in the refresh circuitry itself. This type of failure can only be found by writing a pattern into memory and waiting before checking it for picked or dropped bits. This type of failure also occurs in static chips and can be linked to deficient single cells.

Testing for both types of failure in the same program could result in a test that would take days to run. The philosophy adopted in this program is to provide two separate tests: one that continuously cycles, checking every possible pattern, and one that loads selected patterns, waits and then checks for picked or dropped bits.

Program I/O

Many memory diagnostic programs are without any display or print of the error locations, and I have not seen any that provide a graphic map of the failed locations. In memory troubleshooting, one graphic display is worth a thousand flips of the examine switch. Many of the currently available diagnostics will identify only the first failed location that it sees. This makes it difficult to determine the extent of the failure, and the extent of the failure often provides the best clue as to the cause.

The philosophy with regard to I/O in this program is to dump the failed locations along with the test byte and the byte read out of memory. A memory map is built to identify any 1K × 1 memory block that has a failure in it. Most memory chips in hobby computers are organized as 1K × 1, 2K × 1, 4K × 1, 8K × 1 or 16K × 1, and the display resulting from the map built will tell the story about the extent of any failure.

```
1=LOAD ALL "0" & CHECK
2=LOAD ALL "1" & CHECK
3=MANUAL BYTE LOAD & CHECK
4=RECHECK
5=RECYCLING CHECK
6=PRINT MEMORY MAP
8=REINITIALIZE
9=RETURN TO MONITOR
>1
```

Fig. 2. Input selection display.

A single glance will tell you if the failure is in a single chip or if it falls in all the chips associated with specific addresses. You can even tell if you left a portion of memory in a protected state or if you have left a gap in your RAM after setting your memory address switches. If you want to take the time to personalize the program, it should be able to tell you what chip on which card is bad.

The Program

The program has eight input options implemented in the current version. Table 1 lists the options and their meaning. Fig. 1 shows the initialization input display that allows the memory test boundaries to be set. Note that the top boundary is set using only the high byte of the top address. The top byte location will not be tested, so if you input a 9FH as the test memory top byte, the last byte to be tested will be 9EFFH. Fig. 2 shows the input selection display with test 1 selected (> is the input prompt character).

Option 1 loads all 0s into the memory test area, waits for a short period and then checks the memory for all 0s. Each location that does not contain all 0s during the check prints out the failed location with the test byte and the byte read from the bad location as shown in Fig. 3. Table 2 explains the printout format. At the same time that the errors are being printed, the computer is building the memory map of all the failures. This map is displayed by exercising option 6, but more about that later. After all of the failed locations have been displayed, the option select display is returned and any other option may be selected.

Option 2 is a repeat of option 1 except that all 1s are loaded and checked.

Option 3 allows the insertion and check of any test byte. The display that provides test byte insertion is shown in Fig. 4.

Option 4 allows you to recheck the results of the last test you performed. This allows you to come back hours after you inserted and checked a bit pattern and recheck that same memory load for any changes. (Note: You must run both option 1 and option 2 to check for both picked and dropped bits.)

Option 5 puts you into a recycling test mode that writes patterns into all test locations, checks all locations and then modifies the pattern and checks the new pattern. This check continues until an error is found. Each error is displayed as it occurs in the Table 2 format. The test resumes after displaying the error. The only way to stop this test is to halt the program or reset. In the current configuration the memory map is built but cannot be displayed because the map is zeroed out when the program is reinitialized after the halt or reset. The map is not needed in this test mode because this test is used as a confidence check and/or a way to detect isolated failures. If significant failures are found, run tests 1, 2 or 3 to determine extent.

Option 6 displays the memory map of failed locations generated by tests 1, 2, 3 and 4. The map of a good memory is shown in Fig. 5. Note that it just fits in a 64 character display and that is why the horizontal scale is not marked. Each row of !s indicates a 1 bit by 64K portion of memory. Each ! within a row is a 1 bit by 1K portion of memory. Any bad bit in that 1K block will cause the ! to be converted to a B for bad. Each row corresponds to a specific data bit; the bottom row is data bit D0, and the top row is data bit D7. Fig. 6 shows that there are at least four bad bits in block 9CH. Remember that you need to run both test 1 and 2 to determine the full extent of the failure. The map combined with the failure location printout should enable you to go right to the failed chip or tell you to unprotect that block of memory you forgot about.

ADDRESS BYTE		DATA BYTE	
HIGH	LOW	TEST	BAD
H9F	LFF	TOO	MFF

Table 2. Failed memory byte display format.

```
1=LOAD ALL "0" & CHECK
2=LOAD ALL "1" & CHECK
3=MANUAL BYTE LOAD & CHECK
4=RECHECK
5=RECYCLING CHECK
6=PRINT MEMORY MAP
8=REINITIALIZE
9=RETURN TO MONITOR
>1
H9F LEF TOO MFF
H9F LF0 TOO MFF
H9F LF1 TOO MFF
H9F LF2 TOO MFF
H9F LF3 TOO MFF
H9F LF4 TOO MFF
H9F LF5 TOO MFF
H9F LF6 TOO MFF
H9F LF7 TOO MFF
H9F LF8 TOO MFF
H9F LF9 TOO MFF
H9F LFA TOO MFF
H9F LFB TOO MFF
H9F LFC TOO MFF
H9F LFD TOO MFF
H9F LFE TOO MFF
H9F LFF TOO MFF
```

Fig. 3. Failed memory locations.

```
1=LOAD ALL "0" & CHECK
2=LOAD ALL "1" & CHECK
3=MANUAL BYTE LOAD & CHECK
4=RECHECK
5=RECYCLING CHECK
6=PRINT MEMORY MAP
8=REINITIALIZE
9=RETURN TO MONITOR
>3
INPUT TEST BYTE>3>3
H9F LEF T33 MFF
H9F LF0 T33 MFF
H9F LF1 T33 MFF
H9F LF2 T33 MFF
H9F LF3 T33 MFF
H9F LF4 T33 MFF
H9F LF5 T33 MFF
H9F LF6 T33 MFF
H9F LF7 T33 MFF
H9F LF8 T33 MFF
H9F LF9 T33 MFF
H9F LFA T33 MFF
H9F LFB T33 MFF
H9F LFC T33 MFF
H9F LFD T33 MFF
H9F LFE T33 MFF
H9F LFF T33 MFF
```

Fig. 4. Test byte load and check.

```
1=LOAD ALL "0" & CHECK
2=LOAD ALL "1" & CHECK
3=MANUAL BYTE LOAD & CHECK
4=RECHECK
5=RECYCLING CHECK
6=PRINT MEMORY MAP
8=REINITIALIZE
9=RETURN TO MONITOR
>6
C000111122223333444455556666777788889999AAAABBBBCCCCDDDDEEEEFFFF
C48CC48C048C048C048C048C048C048C048C048C048C048C048C048C048C048C

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Fig. 5. Map of good memory.

```
1=LOAD ALL "0" & CHECK
2=LOAD ALL "1" & CHECK
3=MANUAL BYTE LOAD & CHECK
4=RECHECK
5=RECYCLING CHECK
6=PRINT MEMORY MAP
8=REINITIALIZE
9=RETURN TO MONITOR
>6
C000111122223333444455556666777788889999AAAABBBBCCCCDDDDEEEEFFFF
C48CC48C048C048C048C048C048C048C048C048C048C048C048C048C048C048C

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!B!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!B!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!B!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!B!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Fig. 6. Map of bad memory.

Option 8 allows you to reestablish the memory boundaries.

Option 9 will return you to your system monitor provided that you have placed the address of your monitor in the MON equate statement in line 0430.

Technical Details

The program requires 1K of RAM or EPROM for the basic program. The stack, memory map and variable storage require 256 bytes of good RAM for operation. The location of the 256 bytes of RAM is set in the equate statement MEMS in line 0210. All other RAM locations are referenced to this one so it is the only statement in the source code that must be changed to move the RAM location.

The main program should reside in EPROM ready for use at any point in time that you think that you have a memory problem.

The I/O port and mask configurations are set with equates at the beginning of the listing. Note: The I/O routines are set for active status bit = 1; if your system uses active status bit = 0, you will have to change the JZ *in* the input and output routines to JNZ (lines 2380 and 3940).

I have also inserted my printer form-feed character as an equate in line 0440 so that you can change it if it does strange things to your I/O device. The cursor home and screen clear routine in line 4040 may also cause some trouble. Insert null characters if you have problems. ■

Memory Test program.

```
C000              0010 ********************************
C000              0020 *      MEMORY TEST PROGRAM     *
C000              0030 *    VERSION 5.0 25 MAR 79     *
C000              0040 *        JOHN STANTON          *
C000              0050 ********************************
C000              0060 *
C000              0070 * INPUT OPTIONS
C000              0080 *
C000              0090 * 1=LOAD ALL '0' & CHECK
C000              0100 * 2=LOAD ALL '1' & CHECK
C000              0110 * 3=MANUAL BYTE LOAD & CHECK
C000              0120 * 4=RECHECK
C000              0130 * 5=RECYCLING CHECK
C000              0140 * 6=MEMORY MAP & CHECK
C000              0150 * 8= REINITIALIZE
C000              0160 * 9=RETURN TO MONITOR
C000              0170 *
C000              0180 * SET RAM START HERE - 256 BYTES REQUIRED
C000              0190 * STARTING ON ANY 256 BYTE BOUNDARY (00H)
C000              0200 ****************************************
C000              0210 MEMS   EQU   0DC00H        RAM START LOC
C000              0220 ****************************************
C000              0230 * RAM LOCATIONS - COMPUTED - DO NOT CHANGE!!!
C000              0240 *
C000              0250 BYTE   EQU   MEMS+40H      TEST BYTE LOC
C000              0260 TOP    EQU   MEMS+41H      MEMTOP BYTE LOC
C000              0270 START  EQU   MEMS+42H      HIGH START BYTE LOC
C000              0280 STACK  EQU   MEMS+0FFH     STACK LOC
C000              0290 *
C000              0300 * I/O PORTS - CHANGE IF NECESSARY
C000              0310 *
C000              0320 STAT   EQU   00H           STATUS PORT
C000              0330 PORTI  EQU   01H           INPUT PORT
C000              0340 PORTO  EQU   01H           OUTPUT PORT
C000              0350 *
C000              0360 * I/O PORT MASKS - CHANGE IF NECESSARY
C000              0370 *
C000              0380 MASKI  EQU   01H           INPUT PORT MASK
C000              0390 MASKO  EQU   02H           OUTPUT PORT MASK
C000              0400 *
C000              0410 * MISC EQUATES
C000              0420 *
C000              0430 MON    EQU   0FD00H        SYSTEM MONITOR
C000              0440 FORMF  EQU   0CH           FORM FEED CHAR
C000              0450 *
C000              0460 *
C000              0470 ****************************************
C000              0480 * PROGRAM START                        *
C000              0490 ****************************************
C000              0500 *
C000              0510 *
C000              0520 * INITIALIZATION ROUTINE
C000              0530 *
C000 31 FF DC     0540 INIT   LXI   SP,STACK      LOAD STACK POINTER
C003 CD 54 02     0550        CALL  RUN           LOADS START & STOP LOCATIONS
C006 CD E3 01     0560        CALL  ZERO          ZERO MAP
C009              0570 *
C009              0580 * BEGIN OPTION SELECTION
C009              0590 *
C009 CD 18 02     0600 BEGIN  CALL  CURS          HOME & CLEAR
C00C CD 29 02     0610        CALL  TFORM         SET PRINTER TO TOP OF FORM
C00F CD A9 02     0620        CALL  MENU          PRINT MENU
C012 CD 23 02     0630        CALL  PROMT         PRINT PROMPT
C015 CD 08 02     0640        CALL  INPUT         MAKE SELECTION & ECHO
C018 F5           0650        PUSH  PSW
C019 CD 13 01     0660        CALL  CRLF
C01C F1           0670        POP   PSW
C01D FE 31        0680        CPI   31H           COMPARE FOR 1
C01F CA 4D 00     0690        JZ    TEST1         JUMP TO TEST 1
C022 FE 32        0700        CPI   32H           COMPARE FOR 2
C024 CA 56 00     0710        JZ    TEST2         JUMP TO TEST 2
C027 FE 33        0720        CPI   33H           COMPARE FOR 3
C029 CA 5F 00     0730        JZ    TEST3         JUMP TO TEST 3
C02C FE 34        0740        CPI   34H           COMPARE FOR 4
C02E CA 68 00     0750        JZ    TEST4         JUMP TO TEST 4
C031 FE 35        0760        CPI   35H           COMPARE FOR 5
C033 CA 6E 00     0770        JZ    TEST5         JUMP TO TEST 5
C036 FE 36        0780        CPI   36H           COMPARE FOR 6
C038 CA 71 00     0790        JZ    TEST6         JUMP TO TEST 6
C03B FE 37        0800        CPI   37H
C03D CA 4A 00     0810        JZ    END
C040 FE 38        0820        CPI   38H
C042 CA 0C 00     0830        JZ    INIT          JUMP TO INITIALIZE
```

```
C045 FE 39                     CPI    39H          COMPARE FOR 9
C047 CA 00 FD                  JZ     MON          JUMP TO MONITOR
C04A C3 23 02         END      JMP    PROMT
C04D                  *
C04D                  * MEMORY TEST CONTROL ROUTINE
C04D                  *
C04D CD 7A 00         TEST1    CALL   LOAD1
C05C CD 85 00                  CALL   MTEST
C053 C3 C9 00                  JMP    BEGIN
C056 CD 7F 00         TEST2    CALL   LOAD2
C059 CD 85 00                  CALL   MTEST
C05C C3 09 00                  JMP    BEGIN
C05F CD 93 02         TEST3    CALL   LBYTE
C062 CD 85 00                  CALL   MTEST
C065 C3 09 00                  JMP    BEGIN
C068 CD A8 00         TEST4    CALL   TEST
C068 C3 09 00                  JMP    BEGIN
C06E C3 8E 00         TEST5    JMP    MEMT
C071 CD 66 01         TEST6    CALL   MAPC         MAP CONTROL ROUTINE
C074 CD E3 01                  CALL   ZERO         ZERO MAP
C077 C3 09 00                  JMP    BEGIN
C07A                  *
C07A                  * TEST BYTE LOAD ROUTINE
C07A                  *
C07A 3E 00            LOAD1    MVI    A,00H        LOAD 0
C07C C3 81 00                  JMP    LOAD
C07F 3E FF            LOAD2    MVI    A,0FFH       LOAD FF
C081 32 40 DC         LOAD     STA    BYTE
C084 C9                        RET
C085                  *
C085                  * MEMORY LOAD & TEST ROUTINE
C085                  *
C085 CD 8F 00         MTEST    CALL   MLOAD
C088 CD 9F 00                  CALL   TIMER
C08B CD A8 00                  CALL   TEST
C08E C9                        RET
C08F                  *
C08F                  * MEMORY LOAD ROUTINE
C08F                  *
C08F 2A 42 DC         MLOAD    LHLD   START        LOAD START LOC
C092 3A 40 DC         LOOP1    LDA    BYTE         LOAD TEST BYTE INTO ACC
C095 77                        MOV    M,A          LOAD MEMORY WITH TEST BYTE
C096 23                        INX    H            INCREMENT HREG
C097 3A 41 DC                  LDA    TOP          MOV MEMTOP TO ACC
C09A BC                        CMP    H            COMPARE FOR TOP OF MEMORY
C09B C2 92 00                  JNZ    LOOP1        LOOP IF NOT TOP
C09E C9                        RET
C09F                  *
C09F                  * TIMER ROUTINE
C09F                  *
C09F 11 FF FF         TIMER    LXI    D,0FFFFH     LOAD FFFF IN DEE
C0A2 1D               LOOP2    DCR    E            DECREMENT EREG
C0A3 C2 A2 00                  JNZ    LOOP2        JUMP IF NOT 0
C0A6 15                        DCR    D            DECREMENT DREG
C0A7 C2 A2 00                  JNZ    LOOP2        JUMP IF NOT 0
C0AA C9                        RET
C0A8                  *
C0A8                  * MEMORY TEST ROUTINE
C0A8                  *
C0A8 2A 42 DC         TEST     LHLD   START        LOAD START LOC
C0AE 3A 40 DC         LOOP3    LDA    BYTE         LOAD TEST BYTE INTO ACC
C0B1 BE                        CMP    M            COMPARE MEMORY WITH ACC
C0B2 C4 E6 00                  CNZ    EXIT         EXIT IF ERROR
C0B5                  ** EXITS WITH:
C0B5                  ** A= TEST BYTE
C0B5                  ** BEC= EMPTY
C0B5                  ** HEL= BAD BYTE LOC
C0B5                  ** DEE= EMPTY
C0B5 23                        INX    H            INCREMENT HEL
C0B6 3A 41 DC                  LDA    TOP          MOVE MEMTOP TO ACC
C0B9 BC                        CMP    H            COMPARE FOR TOP OF MEMORY
C0BA C2 AE 00                  JNZ    LOOP3        JUMP IF NOT END OF MEMORY
C0BD C9                        RET
C0BE                  *
C0BE                  * RECYCLING MEMORY TEST ROUTINE
C0BE                  *
C0BE C6 00            MEMT     MVI    B,0          ZERO BREG
C0C0 2A 42 DC         LOOP     LHLD   START        LOAD START LOC
C0C3 7D               FILL     MOV    A,L          LOAD LOW ADD BYTE INTO ACC
C0C4 AC                        XRA    H            EXOR WITH HIGH BYTE
C0C5 A8                        XRA    B            EXOR WITH MODIFIER
C0C6 77                        MOV    M,A          MOVE TEST BYTE TO MEMORY
C0C7 23                        INX    H            INCREMENT ADD
C0C8 3A 41 DC                  LDA    TOP          MOV MEMTOP TO ACC
C0CB BC                        CMP    H            COMPARE FOR MEMTOP
C0CC C2 C3 00                  JNZ    FILL         JUMP IF NOT TOP OF MEMORY
```

```
C142 CD 46 01                  CALL   CONV         CONVERT TO ASCII
C145 C9                        RET
C146                  *
C146                  * BINARY TO HEX CONVERT ROUTINE
C146                  *
C146 78               CONV     MOV    A,B          MOVE ADD TO ACC
C147 1F                        RAR
C148 1F                        RAR
C149 1F                        RAR
C14A 1F                        RAR
C14B CD 53 01                  CALL   HEX          CONVERT DIGIT TO ASCII
C14E 78                        MOV    A,B          PUT ORIG BYTE IN ACC
C14F CD 53 01                  CALL   HEX
C152 C9                        RET
C153                  *
C153                  * HEX TO ASCII ROUTINE
C153                  *
C153 C5               HEX      PUSH   B            SAVE B REG
C154 E6 0F                     ANI    0FH          REMOVE FRONT 4 BITS
C156 C6 30                     ADI    30H          ADD ASCII OFSET 30H
C158 FE 3A                     CPI    3AH          COMPARE FOR 0-9
C15A 47                        MOV    B,A          MOVE ACC TO BREG
C15B DA 61 01                  JC     OUT1         JUMP IF 0-9
C15E C6 07                     ADI    07H          ADD ALPHA OFFSET 07H
C160 47                        MOV    B,A          MOVE ACC TO BREG
C161 CD 1E 01         OUT1     CALL   OUT          OUTPUT RESULT
C164 C1                        POP    B            RESTORE BREG
C165 C9                        RET
C166                  *
C166                  * MAP PRINT CONTROL ROUTINE
C166                  *
C166 CD 18 02         MAPC     CALL   CURS         HOME & ERASE
C169 CD 7C C1                  CALL   MAPH         PRINT TOP LINE OF HEADER
C16C CD 9D C1                  CALL   MAPB         PRINT BOTTOM LINE OF HEADER
C16F CD 8E 01                  CALL   MAPP         PRINT MAP
C172 CD 13 01                  CALL   CRLF
C175 CD 23 C2                  CALL   PROMT        OUTPUT PROMPT
C178 CD 08 02                  CALL   INPUT        WAIT FOR ANY CHARACTER
C178 C9                        RET
C17C                  *
C17C                  * MAP HEADER SUBROUTINE TOP
C17C                  *
C17C C6 3C            MAPH     MVI    B,30H
C17E CE 04            OUTF     MVI    C,04H        LOAD CHARACTER CTR
C180 CD 1E 01         L1       CALL   OUT          OUTPUT BREG
C183 CD                        DCR    C            DEC CHARACTER CTR
C184 79                        MOV    A,C          MOVE CREG TO AREG FOR TEST
C185 C2 80 01                  JNZ    L1           JUMP IF NOT 0
C188 C4                        INR    B            INCREMENT B
C189 78                        MOV    A,B          MOVE CHAR TO ACC FOR CHECK
C18A FE 47                     CPI    47H          CHECK FOR ASCII "G"
C18C CA 99 01                  JZ     END1         JUMP TO END IF "G"
C18F FE 3A                     CPI    3AH          CHECK FOR ASCII ':'
C191 C2 7E 01                  JNZ    OUTF         JUMP IF NOT ASCII ':'
C194 C6 41                     MVI    B,41H        LOAD ASCII "A"
C196 C3 7E 01                  JMP    OUTF         JUMP BACK TO LOAD CHAR
C199 CD 13 01         END1     CALL   CRLF
C19C C9                        RET
C19D                  *
C19D                  * MAP HEADER SUBROUTINE BOTTOM
C19D                  *
C19D CE 1C            MAPB     MVI    C,16         LOAD SEQUENCE CTR
C19F C6 30            L2       MVI    B,30H        LOAD ASCII "0"
C1A1 CD 1E 01                  CALL   OUT          OUTPUT 0
C1A4 C6 34                     MVI    B,34H        LOAD ASCII "4"
C1A6 CD 1E 01                  CALL   OUT          OUTPUT 4
C1A9 C6 38                     MVI    B,38H        LOAD ASCII "8"
C1AB CD 1E C1                  CALL   OUT          OUTPUT 8
C1AE C6 43                     MVI    B,43H        LOAD ASCII "C"
C1BC CD 1E 01                  CALL   OUT          OUTPUT C
C1B3 CD                        DCR    C            DEC SEQ CTR
C1B4 C2 9F 01                  JNZ    L2           TEST FOR SEQ CT OF 0
C1B7 CD 13 01                  CALL   CRLF
C1BA CD 13 01                  CALL   CRLF
C1BD C9                        RET
C1BE                  *
C1BE                  * MAP PRINT ROUTINE
C1BE                  *
C1BE CE 08            MAPP     MVI    C,08H        LOAD BIT CTR
C1CC 21 0C DC         MAPL1    LXI    H,MEMS       LOAD M START
C1C3 7E               MAPL2    MOV    A,M
C1C4 17                        RAL
C1C5 77                        MOV    M,A
C1C6 DA CE 01                  JC     RADB
C1C9 C6 21                     MVI    B,21H        LOAD !
C1CB C3 D0 01                  JMP    RADC         JUMP TO OUT
```

```
COCF 2A 42 DC        1700        LHLD  START        LOAD START LOC
COD2 7D              1710 CHECK  MOV   A,L          LOAD LOW ADD BYTE IN ACC
COD3 AC              1720        XRA   H            EXOR WITH HIGH ADD
COD4 A8              1730        XRA   B            EXOR WITH MODIFIER
COD5 BE              1740        CMP   M            COMPARE WITH MEMORY
COD6 C4 E6 00        1750        CNZ   EXIT         EXIT ON ERROR
COD9                 1760 ** EXITS WITH:
OOD9                 1770 ** A= TEST BYTE
OOD9                 1780 ** B= MODIFIER
COD9                 1790 ** C= EMPTY
COD9                 1800 ** H&L= BAD BYTE LOC
OOD9                 1810 ** D&E= EMPTY
OOD9 23              1820        INX   H            INCREMENT ADD
OODA 7C              1830        MOV   A,H          LOAD HIGH ADD BYTE INOACS
OODB 3A 41 DC        1840        LDA   TOP          MOV MEMTOP TO ACC
OODE BC              1850        CMP   H            COMPARE FOR MEMTOP
CODF C2 D2 00        1860        JNZ   CHECK        JUMP IFNOT END OF MEMORY
COE2 04              1870        INR   B            CHANGE PATTERN
COE3 C3 C0 00        1880        JMP   LOOP         KEEP RUNNING
COE6                 1890 *
COE6                 1900 * ERROR EXIT ROUTINE
COE6                 1910 *
COE6 F5              1920 EXIT   PUSH  PSW          SAVE AREG
COE7 C5              1930        PUSH  B
COE8 E5              1940        PUSH  H            SAVE H&L
OOE9 46              1950        MOV   B,M          MOVE BAD BYTE TO BREG
OOEA 4F              1960        MOV   C,A          MOVE TEST BYTE TO CREG
OOEB                 1970 ** CALLS & RETURNS WITH:
COEB                 1980 ** A= N/A
COEB                 1990 ** B= BAD BYTE
COEB                 2000 ** C= TEST BYTE
COEB                 2010 ** H&L= BAD BYTE LOC
OOEB                 2020 ** D&E= EMPTY
COEB CD F2 01        2030        CALL  MAP          BUILD MEMORY MAP
COEE CD FB 00        2040        CALL  ERROR        PRINT BAD LOC
COF1 CD 29 01        2050        CALL  BAD          PRINT GOOD & BAD BYTES
COF4 CD 13 01        2060        CALL  CRLF         OUTPUT CR LF
COF7 E1              2070        POP   H            RESTORE H&L
COF8 C1              2080        POP   B
COF9 F1              2090        POP   PSW          RESTORE A
COFA C9              2100        RET
COFB                 2110 *
COFB                 2120 * HIGH & LOW ADDOPRINT ROUTINE
COFB                 2130 *
COFB C6 48           2140 ERROR  MVI   B,48H        LOAD HREG WITH H
COFD CD 1E 01        2150        CALL  OUT          OUTPUT H
010C 44              2160        MOV   B,H          LOAD HIGH ADD
C101 CD 46 01        2170        CALL  CONV         CONVERT TO ASCII
C104 C6 20           2180        MVI   B,20H        LOAD A SPACE
0106 CD 1E 01        2190        CALL  OUT          OUTPUT SPACE
C109 C6 4C           2200        MVI   B,4CH        LOAD LREG
C10B CD 1E 01        2210        CALL  OUT          OUTPUT L
C10E 45              2220        MOV   B,L          LOAD LOW ADD
C10F CD 46 01        2230        CALL  CONV         CONVERT TO ASCII
C112 C9              2240        RET
C113                 2250 *
C113                 2260 * CRLF ROUTINE
C113                 2270 *
C113 C6 0D           2280 CRLF   MVI   B,0DH        LOAD CR
C115 CD 1E 01        2290        CALL  OUT          OUTPUT CR
0118 C6 0A           2300        MVI   B,0AH        LOAD LF
C11A CD 1E 01        2310        CALL  OUT          OUTPUT LF
C11D C9              2320        RET
C11E                 2330 *
C11E                 2340 * OUTPUT CHARACTER ROUTINE
C11E                 2350 *
C11E DB 00           2360 OUT    IN    STAT         INPUT STATUS WORD
C120 E6 02           2370        ANI   MASKO        MASK OUTPUT STATUS BIT
C122 CA 1E 01        2380        JZ    OUT          JUMP ON NO STATUS BIT
C125 78              2390        MOV   A,B          LOAD ACC
0126 D3 01           2400        OUT   PORTO        OUTPUT BYTE
0128 C9              2410        RET
C129                 2420 *
C129                 2430 * PRINT INPUT VS OUTPUT BYTE
C129                 2440 *
C129 C6 20           2450 BAD    MVI   B,20H        LOAD SPACE
C12B CD 1E 01        2460        CALL  OUT          OUTPUT SPACE
C12E C6 54           2470        MVI   B,54H        LOAD T
C130 CD 1E 01        2480        CALL  OUT          OUTPUT T
C133 41              2490        MOV   B,C
C134 CD 46 01        2500        CALL  CONV         CONVERT TO ASCII
C137 C6 20           2510        MVI   B,20H        LOAD SPACE
C139 CD 1E 01        2520        CALL  OUT          OUTPUT SPACE
C13C C6 4D           2530        MVI   B,4DH        LOAD M FOR MEMORY BYTE
C13E CD 1E 01        2540        CALL  OUT          OUTPUT M
C141 46              2550        MOV   B,M          MOVE BAD BYTE TO ACC
```

```
01CE C6 42           3420 BADB   MVI   B,42H        LOAD B
01DC CD 1E 01        3430 BADC   CALL  OUT          OUTPUT MAP BYTE
C1D3 23              3440        INX   H            INDEX H&L
C1D4 7D              3450        MOV   A,L
C1D5 FE 40           3460        CPI   40H          COMPARE FOR MAP TOP
C1D7 C2 C3 01        3470        JNZ   MAPL2
C1DA CD 13 01        3480        CALL  CRLF
C1DD 0D              3490        DCR   C            DEC BIT CTR
C1DE 79              3500        MOV   A,C          MOVE CREG TO AREG FOR TEST
C1DF C2 C0 01        3510        JNZ   MAPL1
C1E2 C9              3520        RET
C1E3                 3530 *
C1E3                 3540 * ZERO MEMORY MAP
C1E3                 3550 *
C1E3 E5              3560 ZERO   PUSH  H            SAVE H&L
C1E4 21 00 DC        3570        LXI   H,MEMS       LOAD MAP START ADD
01E7 36 0C           3580 MAPL   MVI   M,0          LOAD 0
01E9 23              3590        INX   H            INCREMENT H&L
01EA 7D              3600        MOV   A,L          MOVE LREG TO ACC
01EB FE 40           3610        CPI   40H          COMPARE FOR MAP TOP
01ED C2 E7 01        3620        JNZ   MAPL         LOOP UNTIL ALL 0
C1FC E1              3630        POP   H            RESTORE H&L
01F1 C9              3640        RET
01F2                 3650 *
01F2                 3660 * BAD BYTE MAP LOAD ROUTINE
01F2                 3670 *
C1F2 C5              3680 MAP    PUSH  B
01F3 F5              3690        PUSH  PSW
01F4 E5              3700        PUSH  H
01F5 7C              3710        MOV   A,H          LOAD HIGH ADD OF BAD BYTE
01F6 CF              3720        RRC
01F7 CF              3730        RRC
01F8 E6 3F           3740        ANI   3FH          REMOVE HIGH ORDER BITS
C1FA 11 C0 DC        3750        LXI   D,MEMS       LOAD RAM START LOC
C1FD 5F              3760        MOV   E,A          MOVE LOW ADD TO EREG
C1FE                 3770 ** D&E CONTAINS ADD OF MAP LOC CORRESPONDING
C1FE                 3780 ** TO THE 1K PAGE WHERE THE BAD BYTE IS LOCATED
C1FE EB              3790        XCHG  LOAD         MAP ADD TO H&L
C1FF 79              3800        MOV   A,C          MOVE TEST BYTE TO ACC
C200 A8              3810        XRA   B            XOR A&B
0201 47              3820        MOV   B,A          MOVE BAD BITS TO BREG
0202 B6              3830        ORA   M            OR MEM WITH ACC
0203 77              3840        MOV   M,A          MOVE ACC TO MEM
0204 E1              3850        POP   H
C205 F1              3860        POP   PSW
C206 C1              3870        POP   B
C207 C9              3880        RET
C208                 3890 *
C208                 3900 * INPUT/ECHO ROUTINE
0208                 3910 *
0208 DB 00           3920 INPUT  IN    STAT         INPUT STATUS WORD
C20A E6 C1           3930        ANI   MASKI        MASK INPUT STATUS BIT
020C CA 08 C2        3940        JZ    INPUT        JUMP IF NO INPUT
C20F DB 01           3950        IN    PORTI        INPUT BYTE
0211 47              3960        MOV   B,A          MOVE BYTE TO BREG
0212 F5              3970        PUSH  PSW          SAVE A
0213 CD 1E 01        3980        CALL  OUT          ECHO BYTE
0216 F1              3990        POP   PSW          RESTORE A
0217 C9              4000        RET
C218                 4010 *
C218                 4020 * CURSOR HOME & ERASE ROUTINE
C218                 4030 *
C218 06 13           4040 CURS   MVI   B,13H        LOAD HOME BYTE
C21A CD 1E 01        4050        CALL  OUT          OUTPUT
C21D 06 16           4060        MVI   B,16H        LOAD ERASE BYTE
C21F CD 1E C1        4070        CALL  OUT          OUTPUT
C222 C9              4080        RET
0223                 4090 *
0223                 4100 * PROMPT OUTPUT ROUTINE
0223                 4110 *
0223 C6 3E           4120 PROMT  MVI   B,3EH        LOAD >
0225 CD 1E 01        4130        CALL  OUT          OUTPUT
C228 C9              4140        RET
C229                 4150 *
C229                 4160 * PRINTER TOP OF FORM
0229                 4170 *
C229 06 0C           4180 TFORM  MVI   B,FORMF      LOAD FORM FEED CHAR
C22B CD 1E 01        4190        CALL  OUT          OUTPUT FORM FEED
C22E C9              4200        RET
022F                 4210 *
022F                 4220 * ASCII HEX TO BIN CONV (2 BYTES)
C22F                 4230 *
C22F CD 23 02        4240 ASCII  CALL  PROMT        OUTPUT PROMPT
0232 CD 08 02        4250        CALL  INPUT        GET FIRST BYTE
C235 CD 48 02        4260        CALL  HEXB         CONVERT TO BIN
C238 87              4270        ADD   A
```

```
C239 87                 ADD  A
C23A 87                 ADD  A
C23B 87                 ADD  A
C23C 4F                 MOV  C,A       SAVE IN C
C23D CD 23 02           CALL PROMT     OUTPUT PROMPT
C240 CD 08 02           CALL INPUT     INPUT 2ND BYTE
C243 CD 48 02           CALL HEXB      CONVERT TO BIN
C246 B1                 ORA  C         OR FITH FIRST BYTE
C247 C9                 RET
C248               •
C248               • BYTE CONVERT HEX TO BIN
C248               •
C248 E6 7F         HEXB ANI  7FH       MASK 8TH BIT
C24A D6 30              SUI  30H       START CONVERT
C24C FE 0A              CPI  0AH       TEST FOR ALPHA
C24E FA 53 02           JM   CONT      JUMP ON 0-9
C251 D6 07              SUI  07H       CONVERT A-F
C253 C9            CONT RET
C254               •
C254               • RUN LOAD PARAMETERS
C254               •
C254 21 41 DC      RUN  LXI  H,TOP     LOAD RAM LOC
C257 CD 18 02           CALL CURS      HOME & CLEAR
C25A E5                 PUSH H         SAVE H&L
C25B 21 60 03           LXI  H,MSG10   LOAD TEXT LOC
C25E CD AC 02           CALL TEXT      GET TEXT
C261 E1                 POP  H         RESTORE H&L
C262 CD 23 02           CALL PROMT     OUTPUT PROMPT
C265 CD 2F 02           CALL ASCII     INPUT HEX CONV TO BIN
C268 77                 MOV  M,A       MOVE BIN TO TOP
C269 CD 13 01           CALL CRLF
C26C 23                 INX  H         STEP POINTER
C26D E5                 PUSH H         SAVE H&L
C26E 21 7A 03           LXI  H,MSG11   LOAD TEXT LOC
C271 CD AC 02           CALL TEXT      GET TEXT
C274 E1                 POP  H         RESTORE H&L
C275 CD 23 02           CALL PROMT     OUTPUT PROMPT
C278 CD 2F 02           CALL ASCII     INPUT & CONVERT
C27B 77                 MOV  M,A       MOVE BIN TO START
C27C CD 13 01           CALL CRLF
C27F E5                 PUSH H         SAVE H&L
C280 21 9B 03           LXI  H,MSG12   LOAD TEXT LOC
C283 CD AC 02           CALL TEXT      GET TEXT
C286 E1                 POP  H         RESTORE H&L
C287 CD 23 02           CALL PROMT     OUTPUT PROMPT
C28A CD 2F 02           CALL ASCII     INPUT & CONV
C28D 23                 INX  H         STEP POINTER
C28E 77                 MOV  M,A       MOVE BIN TO START +1
C28F CD 13 01           CALL CRLF
C292 C9                 RET
C293               •
C293               • LOAD TEST BYTE FROM KBD
C293               •
C293 CD 18 02      LBYTE CALL CURS     HOME & ZERO
C296 E5                 PUSH H         SAVE H&L
C297 21 BD 03           LXI  H,MSG13   LOAD TEXT LOC
C29A CD AC 02           CALL TEXT      GET TEXT
C29D E1                 POP  H         RESTORE H&L
C29E CD 2F 02           CALL ASCII     INPUT & CONV
C2A1 21 40 DC           LXI  H,BYTE    RAM LOC OF BYTE
C2A4 77                 MOV  M,A       MOVE TEST BYTE TO RAM
C2A5 CD 13 01           CALL CRLF
C2AB C9                 RET
C2A9               •
C2A9               • PRINT TEXT ROUTINE
C2A9               •
C2A9 21 BB 02      MENU LXI  H,MSG1
C2AC 7E            TEXT MOV  A,M       MOVE CHARACTER TO ACC
C2AD FE 24              CPI  24H
C2AF CA BA 02           JZ   TERM
C2B2 47                 MOV  B,A
C2B3 CD 1E 01           CALL OUT       OUTPUT
C2B6 23                 INX  H         STEP TO NEXT CHARACTER
C2B7 C3 AC 02           JMP  TEXT      JUMP TO TEXT
C2BA C9            TERM RET
C2BB 31 3D 4C 4F 41 MSG1 ASC '1=LOAD ALL "0" & CHECK'
     44 20 41 4C 4C
     20 22 30 22 20
     26 20 43 48 45
     43 4B
C2D1 0D                 DB   0DH
C2D2 0A                 DB   0AH
C2D3 32 3D 4C 4F 41 MSG2 ASC '2=LOAD ALL "1" & CHECK'
     44 20 41 4C 4C
     20 22 31 22 20
     26 20 43 48 45
     43 4B
C2E9 0D                 DB   0DH
C2EA 0A                 DB   0AH
C2EB 33 3D 4D 41 4E MSG3 ASC '3=MANUAL BYTE LOAD & CHECK'
     55 41 4C 20 42
     59 54 45 20 4C
     4F 41 44 20 26
     20 43 48 45 43
     4B
C305 0D                 DB   0DH
C306 0A                 DB   0AH
C307 34 3D 52 45 43 MSG4 ASC '4=RECHECK'
     48 45 43 4B
C310 0D                 DB   0DH
C311 0A                 DB   0AH
C312 35 3D 52 45 43 MSG5 ASC '5=RECYCLING CHECK'
     59 43 4C 49 4E
     47 20 43 48 45
     43 4B
C323 0D                 DB   0DH
C324 0A                 DB   0AH
C325 36 3D 50 52 49 MSG6 ASC '6=PRINT MEMORY MAP'
     4E 54 20 4D 45
     4D 4F 52 59 20
     4D 41 50
C337 0D                 DB   0DH
C338 0A                 DB   0AH
C339 38 3D 52 45 49 MSG8 ASC '8=REINITIALIZE'
     4E 49 54 49 41
     4C 49 5A 45
C347 0D                 DB   0DH
C348 0A                 DB   0AH
C349 39 3D 52 45 54 MSG9 ASC '9=RETURN TO MONITOR'
     55 52 4E 20 54
     4F 20 4D 4F 4E
     49 54 4F 52
C35C 0D                 DB   0DH
C35D 0A                 DB   0AH
C35E 24                 DB   24H
C35F 24                 DB   24H
C360 49 4E 50 55 54 MSG10 ASC 'INPUT MEMORY TOP HEX BYTE'
     20 4D 45 4D 4F
     52 59 20 54 4F
     50 20 48 45 58
     20 42 59 54 45
C379 24                 DB   24H
C37A 49 4E 50 55 54 MSG11 ASC 'INPUT MEMORY TEST START LOW BYTE'
     20 4D 45 4D 4F
     52 59 20 54 45
     53 54 20 53 54
     41 52 54 20 4C
     4F 57 20 42 59
     54 45
C39A 24                 DB   24H
C39B 49 4E 50 55 54 MSG12 ASC 'INPUT MEMORY TEST START HIGH BYTE'
     20 4D 45 4D 4F
     52 59 20 54 45
     53 54 20 53 54
     41 52 54 20 48
     49 47 48 20 42
     59 54 45
C3BC 24                 DB   24H
C3BD 49 4E 50 55 54 MSG13 ASC 'INPUT TEST BYTE'
     20 54 45 53 54
     20 42 59 54 45
C3CC 24                 DB   24H
```

SYMBOL TABLE

```
ASCII  022F  BAD    0129  BADR   01CE  BADC   01D0  BEGIN  0009  BYTE   DC40
CHECK  00D2  CONT   0253  CONV   0146  CRLF   0113  CURS   0218  END    004A
END1   0190  ERROR  00FB  EXIT   00E6  FILL   00C3  FORMF  000C  HEX    0153
HEXB   0248  INIT   0000  INPUT  0208  L1     0180  L2     019F  LBYTE  0293
LOAD   0081  LOAD1  007A  LOAD2  007F  LOOP   00C0  LOOP1  0092  LOOP2  00A2
LOOP3  00AE  MAP    01F2  MAPB   019D  MAPC   0166  MAPH   017C  MAPL   01E7
MAPL1  01C0  MAPL2  01C3  MAPP   01BE  MASK1  0001  MASK0  0002  MEMS   DC00
MEMT   00BE  MENU   02A9  MLOAD  008F  MON    F000  MSG1   02BB  MSG10  0360
MSG11  037A  MSG12  039B  MSG13  03BD  MSG2   02D3  MSG3   02EB  MSG4   0307
MSG5   0312  MSG6   0325  MSG8   0339  MSG9   0349  MTEST  0085  OUT    011E
OUT1   0161  OUTF   017E  PORT1  0001  PORT0  0001  PROMT  0223  RUN    0254
STACK  DCFF  START  DC42  STAT   0000  TERM   02BA  TEST   00AB  TEST1  004D
TEST2  0056  TEST3  005F  TEST4  006B  TEST5  006E  TEST6  0071  TEXT   02AC
TFORM  0239  TIMER  009F  TOP    DC41  ZERO   01E3
```

At last, Speed and Letter Quality combined in one printer!

Introducing the Sanders Media 12/7 Typographic Printer

It took Sanders Technology to combine the speed of dot matrix and ink jet with the quality of a daisy wheel or Selectric Ball. This is the printer that varies its speed according to the quality of type you want. At its fastest speed (200 cps)—with a single pass of the print head—the characters look similar to those of a typical wire matrix printer. But, the quality is better because the dots are more closely spaced.

Typefaces are also produced with two or more passes of the print head for each line of characters. As the number of passes increases, the shape and quality of the characters improve. Single pass typefaces are ideal for producing drafts at high speed, and four pass fonts produce fully-formed characters of typewriter quality.

Intermix typefaces without skipping a beat. Under software control, the Media 12/7, with its built-in Zilog Z80 microprocessor, can intermix a variety of typefaces. Changes in fonts can be accomplished by a single operator command. No balls or daisy wheels to replace. As many as 11 typefaces can be stored in ROM within the printer.

Capable of reproducing signatures in anyone's handwriting (option), the Media 12/7 can also generate proportionally-spaced characters for printed documents and reports.

Media 12/7 control functions. A few simple commands will control a wide variety of text handling functions. The highly sophisticated software resident within the printer relieves the user's software of many routines needed in word processing systems, thereby freeing valuable computer time. The following features are easily controlled by the operator:

Text Format
- Ragged right
- Justification, with or without letter spacing
- Flush right
- Centered Text

Parameter Controls
- Typeface selection
- Form selection
- Line length
- Left margin
- Indentation
- Ribbon usage
- Insert sequence
- Insert character
- Set/clear underline
- Form length
- Word space
- Letter space
- Line height
- Repeat character
- Draft mode

Print Positioning
- Line feed
- Half line feed
- Form feed
- Absolute vertical tab
- Forced leading
- Negative line feed
- Negative half line feed
- Absolute horizontal tab
- Forced escapement

$3899.00 ~~$3994.00~~

Actual Print Samples

Helvesan Draft (one pass font)

HELVESAN DRAFT IS THE DRAFT QUALITY VERSION OF HELVESAN
SPEED OF THE REGULAR TYPEFACE. AS WITH ALL THE DRAFT EQUI

Helvesan Regular (four pass font)

Helvesan Regular is one of the proportional typefaces
high quality font which is ideal for the production of

Helvesan Italic (four pass font)

The Helvesan Italic typeface is part of the Helvesan
also includes Helvesan Regular and Helvesan Draft. The

Options
- Automatic Sheet feed
- Power Driven Forms Tractor
- Roll Paper Feed
- Signature Printing
- Graphics Output

TWX: 710-348-1796

ComputerCity™

A Division of CPU Industries, Inc.
175 Main Street, Dept. K-3, Charlestown, MA 02129
Hours: 10 AM-6 PM (EST) Mon.-Fri. (Sat. till 5)

✓20

To order call Toll free 1-800-343-6522
Massachusetts residents call 617/242-3350
Franchise and dealer inquiries invited
Prices subject to change without notice.

VISA

master charge

✓Reader Service—see page 257

BASF 6106

5.25" FLOPPY DISK DRIVE

- 40 Track, single or double density
- Smaller size. Fit 3, 6106 drives into the space of 2 SA 400 drives
- Requires less power, generates less heat
- Uses ball bearing friction-free head positioner
- Track to track access time: 12 MSEC.
- Uses industry standard interface and power plugs, and mounting points.

ALL THE ABOVE FEATURES AND MORE FOR ONLY

$299.00 ea.

FOR MORE INFORMATION CONTACT:

OTTO ELECTRONICS ✓89

P.O. BOX 3066, PRINCETON, NJ 08540

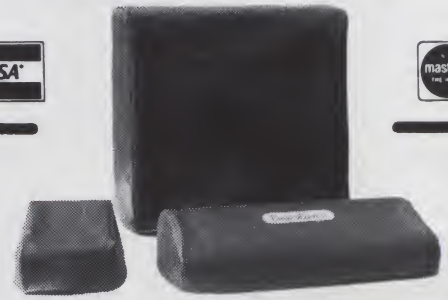
or call **609-448-9165**

MC, VISA, COD accepted NJ residents add 5% sales tax Shipping and insurance extra.

CompuCover®

VISA

master charge



COVER YOUR INVESTMENT

- Cloth Backed Naugahyde Vinyl
- Improved Reliability
- Longer Life
- Waterproof & Dustproof
- Three Decorator Colors
  Saddle Tan • Electra Blue • Black

TRS-80 MODEL II
Entire Unit $22 95
Keyboard Only 7 95

TRS-80 MODEL I
Keyboard $7 95
Cassette 4 95
Video Display 9 95*
Package Offer 19 95*
*Note - Add $3 00 for EXPANSION INTER

Full Apple II
Cover $12 95
Apple II Keyboard 9 95
Apple II Disk 3 95

Pet $15 95
Sorcerer 9 95

Hazeltine
1400, 1410, 1500, 1510, 1520, 1552 $18 95
Soroc IQ 120 18 95
Adds Terminal
980, 100, etc 19 95
ADM-3 14 95
General Automation
3380 19 95
Honeywell VIP
7200 22 95
IBM 5251
Display Station 22 95

Data General
Keyboard $7 95
Data General
Video 15 95

Wang Terminal
without disk $18 95
Wang 2221 Printer 19 95

TRS-80
Line Printer I $17 95
TRS-80
Line Printer II 9 95
Centronics
701, 703 19 95
Centronics 730 9 95
Centronics 779 17 95

T I
800, 810, 820, 825 19 95
Trendcom 100, 200 9 95
Decwriter III
36, 120, 180 19 95
Integral Data Systems
440 12 95

TRS 80 Disk $4 95
Percom Disk 4 95
Double Disk 7 95
Lobo Disk
5¼" mini 6 95
Matchless Disk
5¼" mini 6 95
Vista Disk
5¼" mini 6 95

Send check or money order to:
Include $1.00 for postage and handling.
Overseas orders include $3.00 postage.
DEALER INQUIRES INVITED

CompuCover® ✓90
P.O. Box 324 (Dept. A)
Mary Esther, FL 32569
Phone (904) 243-5793

WHY LOWERCASE?

You can't appreciate just how NECESSARY dualcase is on your TRS-80 until you've seen it. Once you use dualcase you'll never want to go back to UPPERCASE only again. In fact, you'll wonder how you managed without it for so long.

The character generator in your TRS-80 ALREADY contains the following; greek characters, numbers, special symbols, PLUS both UPPER and lowercase letter sets. Block graphics are unaffected. Wouldn't you like access to YOUR entire typeset? Level II Basic converts lowercase command words into UPPERCASE. All characters contained between quotes remain as typed, but the software in an unconverted TRS-80 allows UPPERCASE display only! This software shortcut allowed Tandy to omit one video memory chip. This chip must be added and the video software repaired before the display of dualcase is possible.

Unfortunately,

converting your TRS-80 requires installing the video memory chip plus wiring changes. There is only one modification on the market which eliminates most of the wiring. To get the dualcase mod installed you have three choices: 1) Send your computer to a company or individual who will do the wiring, 2) do it yourself, or 3) "THE PATCH"

To make choices 1 & 2 operate requires using software overhead in the form of a "driver". This takes 30 bytes, unless you want a "normal" shift to UPPERCASE keyboard. That takes upwards of 60 more bytes. Software oriented mods have three more disadvantages: 1) They reside in program memory, eating program space which you could be using, 2) other machine language programs are unusable if they are loaded against the top of memory, or 3) the "driver" software MUST be loaded every time you power-up, or the "MEMORY SIZE?" appears due to program bomb. Choice number three suffers from NONE of the software overhead problems. We call it "THE PATCH" and it's new for the 80's!

"THE PATCH", a small electronic module which plugs into the unused ROM socket on Level II machines, makes necessary software changes to ROM supporting lowercase, an optional block cursor, & extra keyboard debounce. Electronically means NO software overhead. Your computer displays lowercase instantly upon power-up, and the keyboard operates in "normal" typewriter fashion.

"THE PATCH" is completely compatible with your TRS-80 since it is the first, and only, TRS-80 lowercase system designed that flawlessly mates with the computer as a unit, not just a special program package.

"THE PATCH" is also the only modification of any kind which can have extra options and updates factory installed for 5 to 10 dollars per option, as they are available. Same day turnaround.

Order yours today to avoid extra delay. Send $59.97 + 2.50 for S&H to: CECDAT "THE PATCH" / Box 8963 / Moscow, ID 83843.

THE PATCH is a Trademark of CECDAT, Inc.

✓85 master charge VISA

APPLE II PARALLEL INTERFACE CARD

John Bell Engineering is announcing an Apple II Parallel Interface Card. There are four I/O ports with handshaking logic. The board has two 6522 versatile interface adapters and a 74LS74 for addressing and timing. Each 6522 has two interval timers. This will interface your Apple II to printers, speech synthesizers, keyboards, and other John Bell Engineering products. Inputs and outputs are TTL and CMOS compatible.

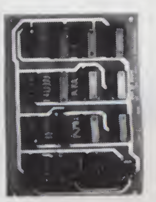
Prices:
79-295 Complete kit $69.95
79-295 Assembled $79.95

SOLID STATE SWITCH

Now you can control the World! Switch lights on and off for home security, computer controlled disco light shows. Turn your printer on only when needed. The Switch can handle 720 watts (120 VAC 6 AMPS). Its input is TTL compatible (5V-2MA), isolation 1500VDC. The circuit board is 2" square on the one channel kit and 2"x 8" on the 4 channel unit.

Prices:
1 Channel kit $9.95 assm. $12.50
4 Channel kit $34.95 assm. $44.95

A to D D to A CONVERTER



John Bell Engineering now has available an Analog to Digital and Digital to Analog Converter Board. Featues low cost medium speed (50,000 conversions per second) for applications such as speech recording and music synthesizing. Single power (+5) required. Parallel inputs and outputs include 8 data bits, strobe lines, and latches. Analog inputs and outputs are medium impedance zero to five volt range.

Prices:
79-287 Kit $49.95
79-287 Assembled $69.95

PRODUCTS AVAILABLE FROM:

JOHN BELL ENGINEERING ✓99
P.O. BOX 338
DEPT. 4
REDWOOD CITY, CA 94064
(415) 367-1137

ADD 6% SALES TAX IN CALIFORNIA AND $1.00 SHIPPING & HANDLING FOR ORDERS LESS THAN $20. ADD 4% FOR VISA OR M.C.

J B ENGINEERING

NEECO

38

Microcomputer Systems Division

PROUDLY ANNOUNCES THE NEWEST HARDWARE AND SOFTWARE FOR YOUR PET!

The PET is now a truly sophisticated Business System with the announcement of these peripherals and software packages.

FREE SOFTWARE & PET DUST COVER! (Offer ends 3/31/80)

NEW

PRODUCT	DESCRIPTION	PRICE	AVAILABILITY
PET 2001—8KN (Large Keys)	8K RAM	$ 795	IMMEDIATE
PET 2001—8K	8K RAM	$ 795	IMMEDIATE
PET 2001—16KN (Large Keys)	16K RAM*	$ 995	IMMEDIATE
PET 2001—32KN (Large Keys)	32K RAM	$1295	IMMEDIATE
PET 2023 PRINTER	ROLL FEED	$ 695	IMMEDIATE
PET 2022 PRINTER	TRACTOR/ROLL	$ 795	IMMEDIATE
ROMRETRO KIT	UPDATED O/S	$ 90	IMMEDIATE
PET 2040	DUAL FLOPPY*	$1295	IMMEDIATE
PET C2N	2nd Cassette	$ 95	IMMEDIATE

*The 16K/32K (large keyboard) units do not include a cassette drive. Order C2N Cassette. 2040 Floppy Drive requires a 16K or 32K unit. 8K RAM Retrofit available July.

ALL PETS ARE FULLY TESTED BY NEECO BEFORE SHIPMENT. NEECO IS A FULL CUSTOMER-ORIENTED COMPANY. CALL FOR OUR FREE CATALOG. ALL ORDERS OVER $795 WILL RECEIVE A FREE NEECO PET DUST COVER AND $100 OF SOFTWARE FROM OUR CATALOG IF YOU MENTION YOU SAW THIS AD.

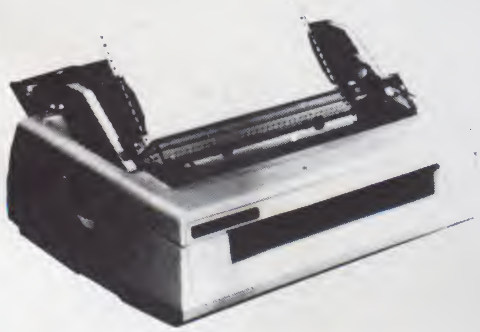
PET-DISK BASED BUSINESS SOFTWARE

SOFTWARE/APPLICATION	*REQUIRES*	*AUTHOR*	*AVAILABILITY*	*PRICE*
WORDPRO II / WORD PROCESSING	2040 + 16K PET	PRO/MICRO	IMMEDIATE	$100
WORDPRO III / WORD PROCESSING	2040 + 32K PET	"	IMMEDIATE	$200
GENERAL LEDGER	"	CMS SOFTWARE	IMMEDIATE	$295*
ACCOUNTS PAYABLE	"	"	IMMEDIATE	$295*
ACCOUNTS RECEIVABLE	"	"	IMMEDIATE	$295*
MAILING LIST	"	"	IMMEDIATE	$100
MICROLEDGER	"	COMPUMAX	"	$140
MICROPAY	"	"	"	$140
MICROREC	"	"	"	$140
MICROINV	"	"	"	$140
MICROPERS	"	"	"	$140

*The CMS Software (G/L, A/R, A/P) are based on Osborne & Associates trial tested business basic software. Software is complete with full documentation and user instructions. All packages require a *printer for output*. Commodore recommends the NEC Spinwriter (available from NEECO) as the output printer for WORDPRO.

PRODUCTS ARE AVAILABLE TO DEALERS VIA MICROAMERICA DISTRIBUTING (617-449-4310)

FOR WORD PROCESSING NEC IS BEST!



* 55 characters per second output speed
* Changeable thimble for different typestyles
* Less than 1% warranty malfunction rate
* IBM quality letter output
* Dealer inquiries invited

THE NEC SPINWRITER
MODEL 5530-P (Centronics I/O modified for PET)

$2995

*Price includes IEEE interface to PET. IEEE Port is available for use with 2040 Dual Disk.

*The NEC 5530-P is the output printer recommended by Commodore for their Word Processing System.

NEECO
679 Highland Ave.
Needham, MA
02194

Mon-Fri 9:30-5:30
MasterCharge & Visa Accepted

(617) 449-1760

MICROAMERICA DISTRIBUTING
"Nationwide distributors of Computer Equipment"
21 Putnam Street
Needham, MA
02194

(617) 449-4310

Audio Feedback for Computer Keyboards

Your typing is sure to improve if you've got rhythm.

George Young
Sierra High School
Tollhouse CA 93667

Watching a skilled typist manipulate the keyboard on a typewriter makes us all envious. If you take the same skilled typist and put him or her on an ASCII keyboard, he or she will begin to make errors. To avoid the errors, the typing speed is dramatically reduced. What is so different about the ASCII keyboard? *It is silent.*

The typist's rhythm is built up with the sound of the keys striking the paper and the platen. In other words, the feedback from the sound of the key striking the paper is used by the skilled typist to trigger the next keystroke on the keyboard. Without realizing it, the typist is using audio feedback, and if this feedback is missing the typist's rhythm is thrown off.

A simple circuit can be added to any ASCII keyboard to provide this missing audio feedback. The audio does not even have to closely resemble the sound of the key striking the paper. It takes only a few minutes to get used to the different sound of the feedback. The important thing is that feedback is present.

Fig. 1 shows this simple circuit. A single chip, the 556, is used to generate the feedback audio. The 556 is a dual timer. Two 555s can just as easily be used to accomplish the same job.

The negative-going keyboard strobe ($\overline{KBS}$) is used to trigger the first half of the 556 (or first 555), which is connected as a one-shot multivibrator. This section of the circuit generates an enable pulse for the second half of the circuit. The second half of the 556 (or the second 555) is connected as an astable multivibrator and generates a tone.

The $\overline{KBS}$ from the ASCII keyboard triggers the first circuit, which generates an enable that turns on the tone oscillator for the length of the enable pulse.

To connect the circuit to your ASCII keyboard you need to find the $\overline{KBS}$ line, +5 volts and ground. The circuit draws so little current that it can be added to the existing +5 supply with no fear of overloading the supply. A small speaker salvaged from a defunct transistor radio can be used for the speaker, or one phone from a pair of headphones can be used to reproduce the audio. The entire circuit takes so little room that it can be tucked away under the keyboard in any available space. No holes need to be drilled in the ASCII keyboard housing as the sound can exit the housing via the space surrounding the keys on the keyboard.

Circuit Fabrication

Just about any fabrication method may be used. The circuit can be wire-wrapped. It can be point-to-point wired. A circuit board can be used. A piece of insulating board such as micarta, glass epoxy or even a piece of ordinary cardboard can be used to hold the chip and the components. Nothing in the circuit is critical, and if component values vary as much as 100 percent from the values shown, the circuit will still work.

The only thing that the beginner need watch carefully is to make certain that the correct pins are connected on the 555 or the 556. This is a linear chip and not as forgiving of wiring errors as are TTL chips. If you have never built anything yet for your computer system, then this would be an excellent first circuit.

Component Values

The beginner usually buys the exact components shown in an author's circuit. Please do not do this. As I have already stated, the component values shown are noncritical. R1 is a pull-up resistor on the KBS line. It may have any value from 1000 Ohms to 10,000 Ohms. We need only make sure that the trigger input (pin 8 on the 556 or pin 2 on the 555) is not left floating.

R2, R3 and C1 in both Fig. 1 and in Fig. 2 determine the enable pulse width, which, in turn, determines the length of the audio tone generated. These three values can be juggled considerably to produce a gate width that gives you a short audio tone each time a key is activated on your keyboard.

R4, R5 and C3 determine the

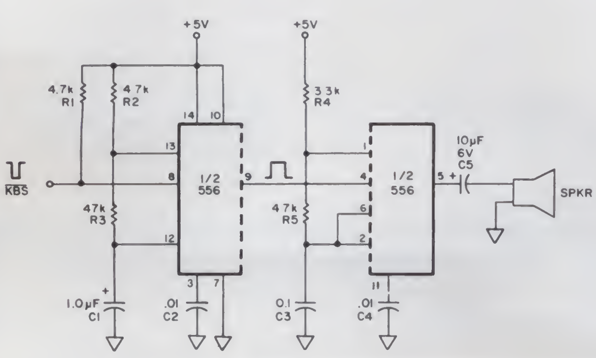


Fig. 1. 556 circuit.

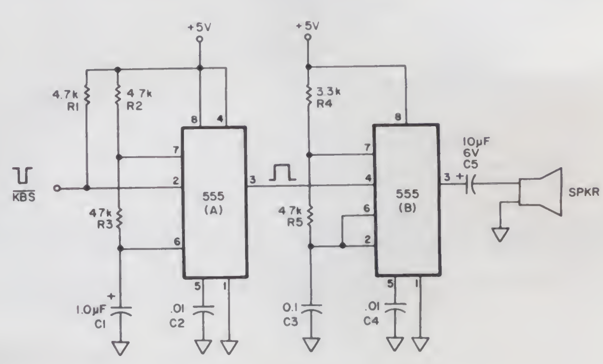


Fig. 2. Dual 555 circuit.

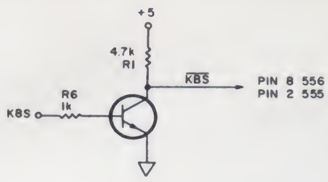


Fig. 3. Transistor inverter for positive keyboard strobe.

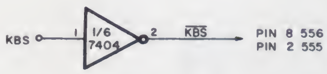


Fig. 4. 7404 inverter for positive keyboard strobe.

tone produced in the speaker. C5 determines the volume of the tone heard in the speaker. Increasing its value makes the tone louder.

Other Potential Problems

You may have a positive-going keyboard strobe (KBS) on your ASCII keyboard. The 556 or the 555 trigger input requires a $\overline{KBS}$. A single NPN transistor (*any* NPN transistor) can be used to invert the KBS to a $\overline{KBS}$. Only the NPN transistor and a current limiting resistor on the base will need to be added. The 4.7k resistor collector load resistor now serves the dual purpose of a collector load and a pull-up on the trigger input of the 556 (or 555). See Fig. 3.

Alternately, one section of a 7404 (or 74LS04) may be used to invert the KBS to $\overline{KBS}$ as shown in Fig. 4. With the TTL inverter, R1 is no longer needed but may be left in the circuit and will not affect circuit operation.

Polled ASCII keyboards (TRS-80, Challenger, IIP, etc.) may be difficult to use with this circuit because they are usually interrupt driven and do not have a KBS or $\overline{KBS}$ available to trigger the circuit. If you don't mind a tone being generated each time your computer system gets an interrupt, the $\overline{\text{Interrupt Request Line}}$ ($\overline{IRQ}$) can be used to trigger the circuit.

Fig. 5 gives a possible PC board layout for the 556 circuit, while Fig. 6 gives a possible PC board layout for the dual 555 circuit. Because the circuit is so simple, it really does not justify the use of a circuit board for its construction.

Both PC board layouts provide a location for the NPN inverting transistor. If you have the required KBS, then simply leave out the transistor. This is not the only use for the circuit. It can be used anywhere you want a tone pulse for any purpose.

I urge you to set up the circuit on your console (see *Kilobaud*, June 1977, p. 78) solderless breadboard and set the pitch of the tone and the duration of the tone to your liking before you execute the circuit in its permanent form. ■

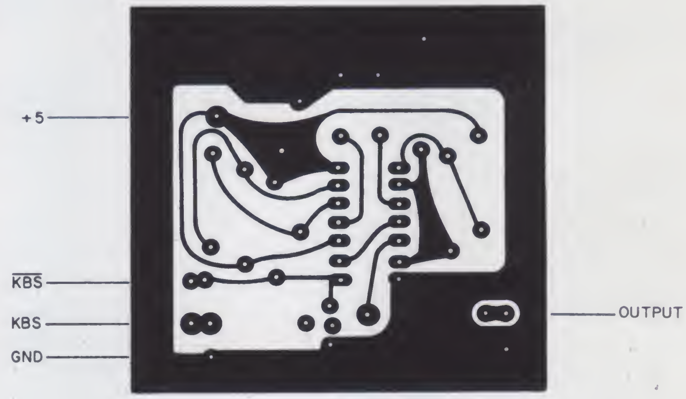


Fig. 5. A possible PC board layout for the 556 circuit.

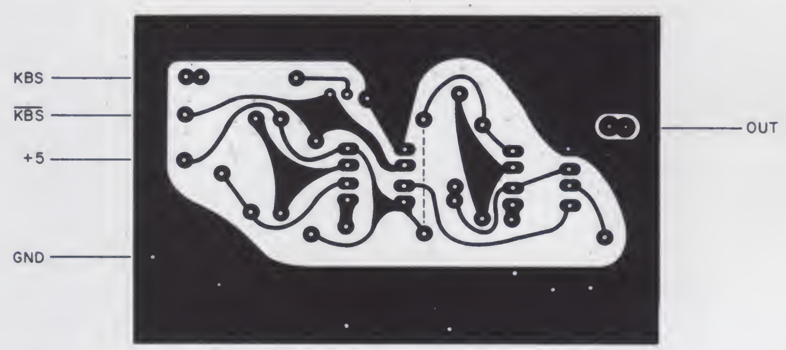


Fig. 6. A possible PC board layout for the dual 555 circuit. The dotted line is a jumper on the component side of the circuit board.

Build a Home for Your Superboard II

OSI's littlest becomes a nice portable package when you house it with a power supply.

Peter G. Hitt
Box 266
La Luz NM 88337

I was not really looking for another computer when Ohio Scientific, Inc., announced the Superboard II in the fall of 1978. I already had a fine Z-80 system operational with lots of memory and a few extra trimmings. However, it lacked a feature that kept gnawing at me: portability. Oh sure, it is movable, but it takes a while to pack it into boxes. It fills the trunk and half the backseat of the car, and then at the other end I have to unload, unpack and set it back up. Am I lazy? Perhaps. But because of its bulk, I cannot easily take it to a friend's house for an evening, and also when I take it to work it is a lot of trouble to bring it home the same day, so it ends up staying the whole week.

Introduction

One day, Mr. S.C. "Doc" Dodd (my boss and fellow engineer) rang my intercom and said, "Better come in here right away—I've got something to show you." Well, as you may have already guessed, he had just received a Superboard II. I was impressed with its neat layout but was nursing some reservations about its usefulness.

During the next two weeks Doc and I spent our spare moments and lunch hours poking and prodding into the intricacies of this compact little computer. Doc added a small circuit (which the board is designed to accept) to drive the office TTY.

By then I was sold on the system for several compelling reasons, primarily, because it is very portable. Also, it has an 8K BASIC-in-ROM, which is rearing to go the moment you power-up. While it is not quite as versatile as some of the 12 to 16K extended BASICs, it has surprisingly complete features and will do about everything that a serious programmer could want.

Finally, the 300 baud KC Standard cassette interface, while slow as cold molasses, is not at all choosy about the quality of its I/O. Doc is using bona fide el-cheapo tape from the local variety store and a $20 recorder, which was literally salvaged from a boys' toy box . . . and he hasn't had a misplaced bit yet!

About this time we were writing BASIC programs, which were ricocheting off the top end of his 4K RAM, so Doc placed an order for an additional 4K and I placed an order for my own Superboard II with 8K. Also, we finally got around to considering how we were going to package the boards so that they would be protected from the rigors of transporting and how to add a built-in power supply. Six weeks later UPS delivered my computer, and after another couple of weeks of wearing the "new" off of it, I began the construction of the case and power supply.

Power Supply

The first order of business was to make a full-scale side view layout of the case showing

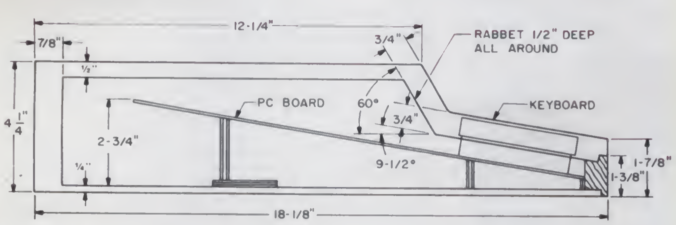


Fig. 1. A cross-section view showing dimensions of the side boards and placement of the computer board.

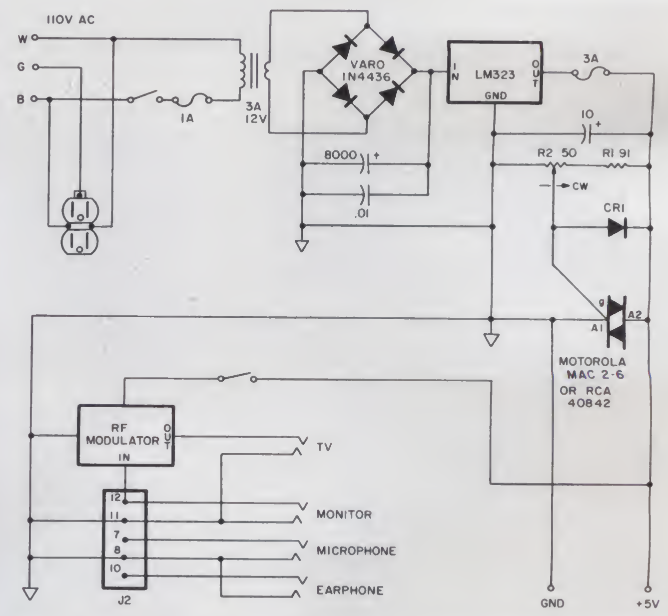


Fig. 2. Diagram of the 5 volt, 3 Amp regulated power supply with crowbar voltage protection. The rf modulator mentioned in the text enables the use of an unmodified TV set as a monitor. Plug J2 is located on the left rear corner of the Superboard II.

how the PC board would fit into it (Fig. 1). Enough space was provided above the board in the event that I might add an expansion board someday. Before the case design could be finalized, however, I had to build the 5 volt, 3 Amp power supply.

As shown in the circuit diagram (Fig. 2), the power supply is simple and concise, using a 3 Amp, 12 volt filament transformer, bridge rectifier, an 8000 uF filter capacitor and an LM 323, 3 Amp, 5 volt voltage regulator. It also includes a clever crowbar circuit that not only protects the computer from excessive positive or negative voltages, but also protects the supply itself by zapping a 3 Amp fuse. This bidirectional crowbar circuit was recently published in one of the trade journals as a design idea submitted by David L. Sporre of Singer-General Precision.

The trigger voltage of most triacs is between 0.6 and 1.5 volts. The adjustable voltage divider (R2 and R1) applies a proportional part of the supply voltage to the gate. To adjust the crowbar threshold, first turn R2 fully counterclockwise to set the gate at ground potential. Attach a variable voltage power supply and voltmeter to the output leads of the crowbar and adjust the voltage to 5.7 volts.

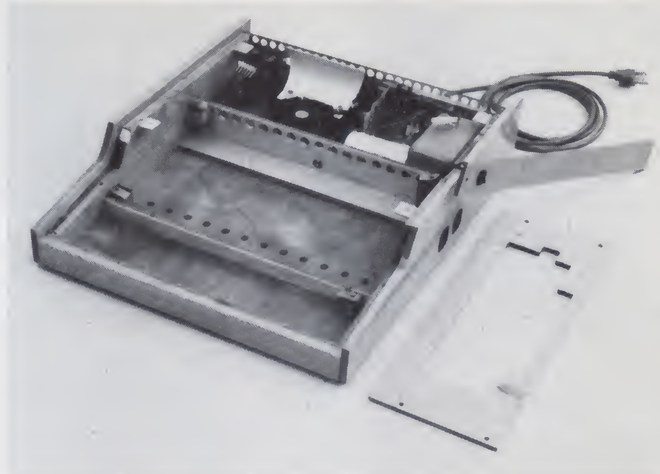


Photo 2. Top view showing construction details of the framework and placement of the power supply and fan. The four small wooden tabs at the ends of the vertical crosspieces serve to clamp the PC board firmly in place.

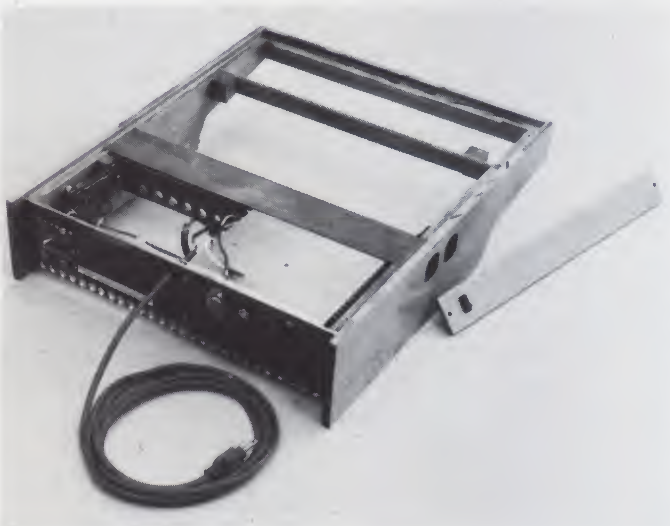


Photo 1. View of the bottom of the case with the bottom panel removed. The crosspieces arranged both horizontally and vertically give the framework rigidity. The bottom panel fits flush into the rabbeted edges. The small circuit board fastened to the left side board is the rf modulator, which permits video output to a normal TV set. (All photographs by Tom Gilmore and Gilbert Montoya)

Next, turn R2 very slowly clockwise until the triac fires. You will know when it fires because the voltmeter will suddenly drop to zero. What has happened is that the triac has shorted the 5.7 volts to ground—so shut off the power supply quickly since the whole idea is to set the trigger level of the triac and not to burn it up! The diode, CR1, serves to protect your computer in case a negative voltage transient should occur. If the +5 lead, for any reason, ever went negative with respect to ground, the crowbar would actuate.

Now it is time to hook up the crowbar to the power supply that you have built and try it out. Caution: Do not attach the supply to the computer at this time because we need to know more about the little beast before doing anything drastic.

Using a suitable ammeter on the output, place a load in series with it across to the ground terminal. Watch the ammeter when you turn on the power supply and increase or decrease the load as necessary to fix the output at 2 Amps (the Superboard is going to draw about 1.8 Amps). Doc used automotive taillight bulbs in series and parallel combinations to trim the load. I borrowed his bulbs to test my supply.

When the amperage is set, place a voltmeter across the load and let it cook for several hours, during which time you should monitor it closely. It should maintain the 2 Amps at 4.8 to 5.2 volts. This test will give you some valuable information about heat buildup, because after an hour or so everything will start to get warm.

If you have followed good design practices and provided adequate heat sinks for the main hot spots, namely the transformer, voltage regulator and bridge, the temperature should level off somewhere short of a "meltdown." My power supply ran hotter than I thought it should (although within the rated limits of the components), so I mounted a 3-inch block fan in the case. The little fan did a more-than-adequate job but was noisy, so Doc put a 10 Watt, 300 Ohm resistor in series with it to slow it down.

I was watching all this and

Photo 3. Compressed into a 4 3/4 × 6 × 3 inch package, the power supply occupies a rear corner of the enclosure. The small perforated board contains a crowbar circuit that protects the computer from excessive voltage. The keyboard panel and switch panel are tried on for fit.

Photo 4. Installation of the Superboard II in the finished case. Cooling air from a small fan is diverted partially forward across the top of the board by a curved cardboard baffle while most of the air is diverted sideways into the power supply.

biting my tongue to keep from protesting out loud. In my mind's eye I could see that old equation, AC motor + low voltage = burn-up. "The worst that could happen," I kept telling myself, "would be the price of a new fan—probably less expensive than contradicting the boss."

An hour later we stopped the fan and I wet my finger before touching it. This was a big anticlimax because it was not even warm. Doc explained that we got away with it because slowing the fan down so greatly reduced the power consumption that there was no danger of its overheating. There was some extra heat thrown off by the 10 Watt resistor, but, as I said, the fan was more than adequate for this cooling job.

Case Construction

After you have thoroughly tested the power supply, you can assume that it is trustworthy enough to hook it up to your computer. Also, you can now proceed with determining the final dimensions of the case. Whether you make the case of wood or metal will depend on which materials you have on hand and which you work with best. I built my case entirely of wood, except for the one-eighth-inch aluminum backplane that serves as a heat sink and a mounting surface for the power supply and the I/O plugs and switches.

Photos 1 and 2 depict the construction details of the framework. The interior dimensions of the case measure 13½ inches wide, 16½ inches long and 3½ inches high. The external dimensions are 15 × 18 × 4¼ inches. The side boards are made of ¾ inch thick maple, as is the crosspiece in front of the keyboard. Both edges and both ends of the side boards are rabbeted to receive the top and bottom panels, front crosspiece and backplane. The top, bottom and keyboard panels are made of ¼ inch cabinet grade maple plywood.

One horizontal and two vertical crosspieces, also made of ¼ inch plywood, serve two purposes. They provide necessary strength and rigidity to the framework and provide support for the PC board. Their placement was determined by the position of transverse corridors on the back of the PC board, which are void of solder-tails, thus allowing the board to rest firmly on these supports.

These plywood crosspieces are set into routed grooves in the side boards, and the assembly is glued together with epoxy cement. Use the top and bottom panels to hold everything square and clamp securely while the epoxy sets. Do not cement the top and bottom panels, as they provide access to the computer and will be held in place with screws. Caution: If you use five minute epoxy, have everything in readiness before you mix the cement because the stuff waits for no man! It would be wise to enlist some help if you plan to use the fast-cure epoxy. After you remove the clamps, you may want to reinforce each corner with some small corner blocks of maple cemented in place.

Next, as shown in Photos 2 and 3, make the keyboard panel of the same ¼ inch cabinet-grade maple plywood. Making a close fitting cutout is challenging, but it can be easier if you first make a thin cardboard template. It will be necessary to mount the PC board temporarily so that you can measure and cut out the template to get an exact fit. Don't be satisfied with anything less than a perfect fit because when your friends see your finished product, the first thing they will look at is the keyboard.

Now transfer the panel outline and cutout pattern to the plywood panel and cut it out carefully using a good, sharp coping saw, jigsaw or, better yet, a router. Remember: Dull saw blades make a mess of plywood, and the damage cannot be repaired or sanded away. Good wood is expensive, so double-check everything before you start sawing. As my grandpappy used to say, "Measure twice and cut once!"

When all the panels are made, give the whole thing a good sanding and two coats of bartop varnish. Sand the first coat lightly with fine sandpaper and wipe it dust free before you put on the second coat. Varnish carefully applied will give you a finish you can be proud of.

The Superboard can now be mounted in the finished case (see Photo 4) and secured with four maple clamps. You can now wire up all power conductors, switches and I/O plugs. The panels and backplane can also be attached to the side boards with screws.

While designing my case I thought that a 110 V ac duplex receptacle would be useful to include. There was no room for it on the backplane, so I placed it in the right-hand side board. This turned out quite well, but I do not recommend doing this unless you have a router, since it must be set into a deep pocket so that the outlets will set flush with the outer surface.

If you have decided to use a small fan, you will need to locate it on the bottom board near the backplane as shown in Photo 2. I had drilled convection holes in the backplane, top panel and bottom panel before I decided to use a fan and I found it neces-

Photo 5. Computing reduced to its simplest terms. While expansion into a full-blown system is possible, the Superboard II really shines in this minimal form with maximum portability and affordability.

sary to install a stiff paper baffle to divert cooling air into the interior of the case and to keep it from short-cutting out the nearest holes. You will also need to install four rubber feet on the bottom panel so that the fan (or convection holes) can intake cool air. The finished computer is shown in Photo 5.

Conclusion

Before ending this description, I will clarify one more part of the circuit diagram (Fig. 2). To enable us to enjoy maximum portability with our computers, Doc and I both decided to add an rf modulator to the video output. The modulator is tuned to a VHF channel not used in our area. With this modulator we do not need to carry a video monitor. You can generally find a TV set wherever you go, and all you need to do is disconnect the external antenna and clip onto the VHF lugs. Plain, inexpensive lamp cord is a good impedance match, and you do not need coaxial cable for this connection.

With this little modulator you now have a usable monitor with no modifications to your friend's TV set. The circuit diagram for this modulator is not included because to do so would violate copyright laws. Several manufacturers sell rf modulator kits, and you can probably find one or two of their advertisements in the pages of this magazine. Another source for the rf modulator could be the adaptation of a modulator from a TV game set.

Some of you readers are probably asking yourselves why I didn't order the Challenger IP computer instead of the Superboard II since it is the identical board, and for just 70 bucks more you get the case and power supply. To that I can only answer that I already had the wood and most of the major power supply components on hand—but mostly, I enjoyed doing it! ■

Budget-Minded A/D Converter

A handful of components and a few routines turn your 8080 machine into a thermometer.

William G. Delinger
Richard L. Petkiewicz
Dept. of Physics
Northern Arizona University
Flagstaff AZ 86011

Analog quantities such as temperature values are usually digitized and interfaced to a microprocessor with commercially available analog-to-digital converters; but even 8-bit converters are relatively expensive. Also, it is often necessary to use a multiplexer to interface several temperature signals. If fast conversion speeds are not of prime importance, however, then you can use software to replace hardware.

The temperature-conversion method we describe here uses a serial-polling technique suggested by Hal Chamberlin ("Try Solar Energy," *Kilobaud*, June 1977, p. 88). This method does not require a multiplexer; nevertheless, it allows up to eight signals to be connected to one 8-bit input port of a microprocessor. In addition, it offers the advantages of being easy to implement and low cost. This approach assumes, of course, that the microprocessor can be dedicated to such tasks for the required amount of time and that the necessary memory is available.

Hardware

The basic idea of the conversion method is illustrated in block-diagram form in Fig. 1. A frequency-modulated square wave is produced by some type of analog-to-frequency converter. For example, we used a thermistor connected as a timing element of a 555-timer oscillator circuit as shown in Fig. 2. Changes in the thermistor's temperature caused corresponding changes in the frequency of the output square wave generated by the oscillator. This square wave was connected to one bit of the microprocessor input port.

We then designed a program stored in the microprocessor memory to continually poll the incoming wave and count the length of time the wave was in the low state (logic 0) during one complete cycle of the wave. We used an interpolation routine to convert the count value into a digital temperature reading.

The thermistor used in the temperature-sensing circuit was a Western Thermistor Corporation type 1C1502-1 with a nominal resistance of 15k Ohms at room temperature (25°C). The resistance-versus-temperature curve for this device is shown in Fig. 3. The thermistor and the 4.7 uF capacitor illustrated in Fig. 2 formed the timing circuit for the square-wave oscillator.

During a complete cycle, the wave produced by the oscillator was in the low state (logic 0) for 0.693 RC seconds, where R is the resistance of the thermistor in Ohms and C is the value of the capacitor in farads. With the values given, the low-state periods of the wave varied from about 150 to 3 milliseconds for temperatures of 0 to 99°C, respectively.

This conversion scheme was implemented on a microprocessor trainer kit originally developed by Gordon D. Jones at Lawrence Livermore Laboratory. The assembled kit is shown in the accompanying photo. It is a self-contained, microprocessor system housed in a briefcase for portability and convenience of use. It utilizes the Intel 8080A microprocessor and associated supporting integrated circuits.

The trainer has 512 bytes of RAM, 768 bytes of PROM, an 8-bit input port and a latched 8-bit output port. The microprocessor system uses a crystal-controlled 1 MHz frequency clock. The programs were entered and executed by means of a hexadecimal keyboard provided with the kit. The two-digit hexadecimal display (output port 0) of the trainer was used for the digital temperature readout. Provisions were made to display temperatures in the range from 0 to 99°C.

Main Program

The main program for this application consists primarily of calls to the various subroutines as shown in the program list-

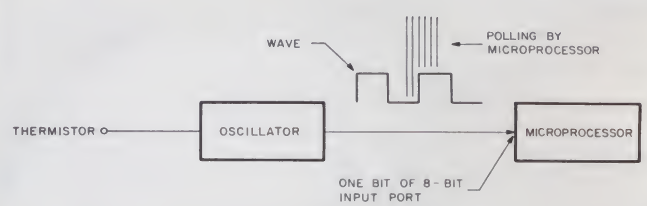
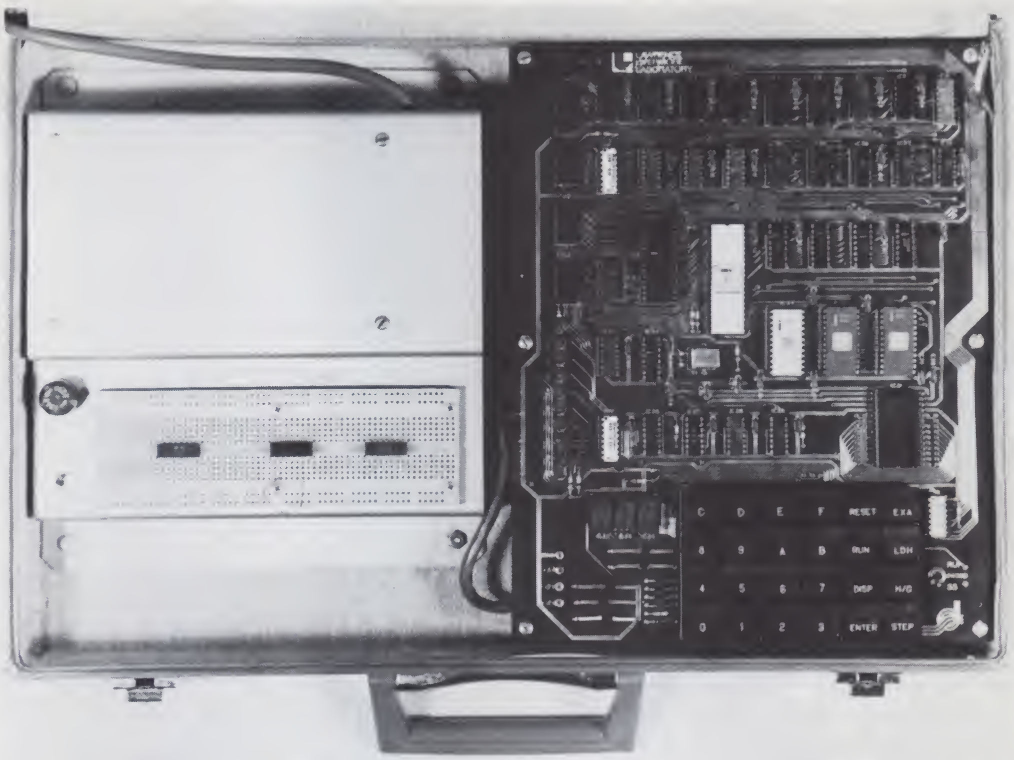


Fig. 1. The microprocessor continually polls the input wave produced by the oscillator. Changes in the temperature of the thermistor cause changes in the frequency of the wave produced by the oscillator.

ing. The program was designed to output the temperature about every second. First, a hexadecimal value of 01 was loaded into the B register. This served the purpose of providing the mask byte to pick out the bit to which the square wave was connected.

The POLL subroutine takes the incoming wave and calculates a double word digital count and stores the result in registers D-E. Then CHECK determines if the temperature is within the range of the tabulated values. If the input is valid, INTER takes this count and interpolates to obtain a temperature. Next, HEX changes the temperature in hexadecimal to a decimal temperature. The temperature is displayed, DELAY causes a 1 second pause and the program jumps back to the beginning to repeat the process.



The assembled 8080A-based microprocessor trainer kit in briefcase.

Subroutines

POLL. The POLL subroutine is the heart of the temperature-conversion method. This routine was written to get high resolution for the period of square wave used. Although both the high state and low state of the oscillator output-wave varied with temperature, the low state was chosen for the counting phase of this routine. Since the microprocessor polling signal can intercept the wave at anytime, we designed the program to start counting only when the square wave made a transition from the high state to the low state.

The first section of code checks to see if the wave is already in the low state. If this condition is true, then the routine will go into a waiting loop until the wave is at the beginning of the high state. If the wave was in the high state initially, the program would fall through to the next waiting loop. Finally, when the high-to-low transition occurs, the counter (registers D-E) is initialized to 1 because the preceding scanning section takes about 1 count. The last part of the routine keeps monitoring the wave until the opposite transition (low to high) signals the end of the counting period.

For every loop completed at this section of code, the counter is incremented by one count until the end of the low state is reached. Each counting loop takes 29 clock periods. Therefore, for a system with a 1 MHz clock frequency, the resolution is 29 microseconds per count. This will give an accuracy of 1 percent for waves with a period of 3 milliseconds. Proportionally higher accuracies can be obtained with longer period waves.

CHECK. During the interpolation, it is possible that the count being interpolated is beyond the range of the table. Before the main program calls the interpolation routine, the subroutine CHECK will determine if the double word count is within the range of the table.

The values CHIGH and CLOW must be negative, or two's complement, of the upper and lower range of the data table, respectively. These limit values are each loaded into the BC registers. A double-word add is performed to see if the count is within the bounds of the table. If it is, the subroutine returns to the main program.

If the count is beyond the range of the table, an error condition is set, the hex value FF is loaded into the A register and this flag indication is displayed. The routine then returns to the main program. The main program jumps back to the beginning to try again. If not, the count is interpolated.

INTER. The interpolation routine was used to take the double word count (16 bits) from the polling routine and perform a linear interpolation to obtain a temperature. The procedure for this routine was derived from a standard algebraic linear interpolation.

For this problem, the count was calculated every ten degrees from 0 to 100°C for the calibrated thermistor. The data

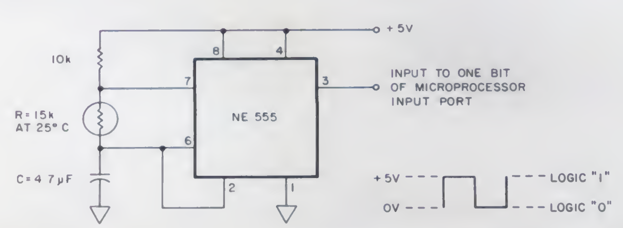


Fig. 2. The square wave produced by the 555-timer oscillator circuit is controlled by the temperature of the thermistor. During a complete cycle, the wave is in the low state (logic 0) for 0.693 RC seconds, where R is the resistance of the thermistor in Ohms and C is the value of the timing capacitor in farads.

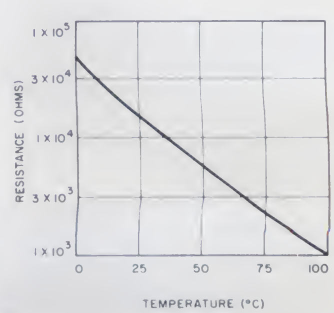


Fig. 3. Logarithm of resistance versus temperature for a Western Thermistor Corporation type 1C1502-1 thermistor.

was stored in a table by use of the 8080 assembly-language define-word command as shown in the program listing.

The values were stored in the following formatting sequence: the first two bytes of the sequence are the negative (two's complement) double word count for the specified temperature; the third byte is the negative (two's complement) of $(C_i - C_{i-1})/10$, where C_i is the count for the i *th* specified temperature and C_{i-1} is the count for the (i – 1) *th*, or next lowest calculated temperature, with i ranging from 1 to 11 for the entire table; the last byte of each four byte sequence is the temperature.

The program starts by loading the start address of the data table into the H-L register pair. Then registers D-E are saved by a push operation because the information in these registers is destroyed by the arithmetic operations. The first segment of code searches the data table by comparing the count in D-E with the table values for the double word count until the proper interval for the count in D-E is found.

Once the interval is found, the interpolation begins. The value FF is loaded into B to extend the sign of C so that B-C is a two's complement double word. Register pair D-E at this point contains the difference between the count that D-E started with and the table value. A pseudo division is performed on D-E by B-C (which contains $(C_i - C_{i-1})/10$) by continuous addition of the negative value B-C and simultaneous incrementing of a counter (A) until D-E is negative. No rounding occurs, so the result is truncated.

This value is now subtracted from the temperature in the table to obtain the temperature corresponding to the original value of D-E. Now D-E is returned to its original value by popping it off of the stack. Its corresponding temperature is returned in A.

HEX. This routine takes a hexadecimal number in register D and converts it to a binary coded decimal (BCD) number. Two counters, B and C, are set up. Register C counts in hexadecimal and register B counts simultaneously in BCD. When registers D and C are equal, then register B will have the BCD value corresponding to the hexadecimal value that was in the register D.

The BCD count for register B is accomplished by using the DAA command. The value in B is moved to register A after both registers have been incremented. Then, a decimal adjust is performed on register A. This will cause the hexadecimal value of 0A to be converted to 10, 1A to be converted to 20, and so on. This routine returns with the decimal number in register D.

DELAY. This routine is just used to cause a pause so the temperature can be displayed. Register B is set to zero. Then B is incremented until it resets to zero; this occurs every 256 counts. Register C is decremented every time B is reset. This causes a total of 65,536 loops, or approximately a one second delay.

Conclusion

The routines explained in this article can be applied in other applications. For example, the polling routine could be used to convert analog light levels to digital values if a light-dependent resistor is connected in place of the thermistor in Fig. 2. Similarly, other types of transducers could be substituted. Also, by changing the mask byte (register B) of the polling routine, any of the eight bits of input port 1 can be monitored. A main program could be written as a software multiplexer to successively monitor each of the eight bits and output the result. Or, up to eight different signals could be monitored in any desired sequence.

In a like manner, the interpolation routine can interpolate other types of data. It should be remembered, however, that the data must follow some restrictions. For instance, the data should be evenly spaced and the count values should be stored in two's complement form as discussed in the INTER subroutine section of this article. Also, the interpolation is a linear interpolation program. If the function of the digital count is highly nonlinear, then closely spaced points must be used, or a more sophisticated interpolation routine should be written.

The price for this analog-to-digital converter is reasonable. At present, for instance, the NE 555 timer costs about 50 cents. Thermistors that will work properly in the circuit of Fig. 2 can be purchased for about $3 each. ■

References

1. J. P. Bauernschub, Jr., "Temperature Sensing," *Kilobaud*, March 1978, pp. 110–118.
2. H. M. Berlin, *The 555 Timer Applications Sourcebook with Experiments*, E & L Instruments, Derby CT, 1976.

Program listing. The 8080 assembly-language listing of the analog-to-digital temperature conversion program.

```
                ;*********************************************
                ;*                                           *
                ;*  TEMPERATURE MONITORING PROGRAM           *
                ;*                                           *
                ;*  MAIN PROGRAM                             *
                ;*                                           *
                ;*********************************************
0600                    ORG     0600H
0600  31F007            LXI     SP,07F0H  ;INITIALIZE STACK POINTER
0603  CD4806    ST:     CALL    DELAY     ;PAUSE
0606  0601              MVI     B,01H     ;SET MASK BYTE FOR POLL
0608  CD1E06            CALL    POLL      ;DO POLLING
060B  CD5506            CALL    CHECK     ;CHECK THE COUNT
060E  FEFF              CPI     0FFH      ;CHECK FOR AN ERROR
0610  CA0306            JZ      ST
0613  CD7406            CALL    INTER     ;DO INTERPOLATION
0616  CD3506            CALL    HEX       ;CHANGE HEX TO DECIMAL
0619  D300              OUT     0         ;DISPLAY TEMPERATURE
061B  C30306            JMP     ST        ;REPEAT
                ;*********************************************
                ;*                                           *
                ;*  PROGRAM:  ANALOG/DIGITAL POLLING ROUTINE *
                ;*                                           *
                ;*********************************************
061E  DB01      POLL:   IN      1         ;READ INPUT PORT 1
0620  A0                ANA     B         ;MASK FOR DESIRED BIT
0621  CA1E06            JZ      POLL      ;KEEP LOOPING IF BIT=0
0624  DB01      HIGH:   IN      1         ;READ INPUT PORT 1 AGAIN
0626  A0                ANA     B         ;MASK FOR DESIRED BIT
0627  C22406            JNZ     HIGH      ;KEEP LOOPING IF BIT=1
062A  110100            LXI     D,0001H   ;INITIALIZE COUNTER
062D  DB01      LOW:    IN      1         ;READ INPUT PORT 1 AGAIN
062F  A0                ANA     B         ;MASK FOR DESIRED BIT
0630  13                INX     D         ;INCREMENT COUNTER
0631  CA2D06            JZ      LOW       ;KEEP COUNTING IF BIT=0
0634  C9                RET               ;RETURN COUNT IN D&E
                ;*********************************************
                ;*                                           *
                ;*  PROGRAM:  HEX TO DECIMAL                 *
                ;*                                           *
                ;*********************************************
0635  0600      HEX:    MVI     B,0       ;SET UP 2 COUNTERS, ONE
0637  0E00              MVI     C,0       ;WILL COUNT IN HEX (C)
0639  7A        START:  MOV     A,D       :THE OTHER IN DECIMAL (B)
063A  B9                CMP     C         ;COMPARE C AND D
063B  CA4606            JZ      SAVE      ;WHEN EQUAL, RETURN.
063E  0C                INR     C         ;INCREMENT BOTH REGISTERS
063F  04                INR     B
0640  78                MOV     A,B
0641  27                DAA               ;DECIMAL ADJUST B
0642  47                MOV     B,A
0643  C33906            JMP     START
0646  78        SAVE:   MOV     A,B       ;RETURN WITH THE DECIMAL
0647  C9                RET               ;NUMBER IN A
                ;*********************************************
                ;*                                           *
                ;*  PROGRAM:  DELAY                          *
                ;*                                           *
                ;*********************************************
0648  0600      DELAY:  MVI     B,0       ;INITIALIZE B AND C
064A  0EFF              MVI     C,0FFH
064C  04        LOOP:   INR     B         ;INCREMENT B UNTIL IT IS
064D  C24C06            JNZ     LOOP      ;RESET TO 0
0650  0D                DCR     C         ;DECREMENT C EACH TIME
0651  C24C06            JNZ     LOOP      ;B IS RESET
0654  C9                RET
                ;*********************************************
                ;*                                           *
                ;*  PROGRAM:  CHECK                          *
                ;*                                           *
                ;*********************************************
0655  219D06    CHECK:  LXI     H,CHIGH   ;LOAD THE HIGH VALUE
0658  4E                MOV     C,M       ;INTO BC.
0659  2C                INR     L
065A  46                MOV     B,M
065B  D5                PUSH    D
065C  EB                XCHG              ;ADD. IF THERE IS A
065D  09                DAD     B         ;CARRY, THEN
065E  DA6E06            JC      ERR       ;DE > CHIGH.
0661  D1                POP     D
0662  219F06            LXI     H,CLOW    ;LOAD THE LOW VALUE
0665  4E                MOV     C,M       ;INTO BC.
0666  2C                INR     L
0667  46                MOV     B,M
0668  D5                PUSH    D
```

3. H. Chamberlin, "Try Solar Energy," *Kilbaud*, June 1977, pp. 88–91.

4. S. D. Conte, *Elementary Numerical Analysis*, McGraw-Hill, New York, 1965.

5. W. G. Delinger and R. L. Petkiewicz, "Analog/Digital Polling Routine," Intel User's Library Program D46, August 1977.

6. Intel Corporation, *8080/8085 Assembly Language Programming Manual*, 3065 Bowers Ave., Santa Clara CA, 1978.

7. G. D. Jones, *MST-80 Microprocessor Trainer Manual*, Lawrence Livermore Laboratory Report UCID-17145, Livermore CA, 1976.

8. T. A. Seim, "Numerical Interpolation for Microprocessor-Based Systems," *Computer Design*, Feb. 1978, pp. 111–116.

9. Western Thermistor Corporation, "Thermistors," 354 Via Del Monte, Oceanside CA, 1977.

```
0669   EB                XCHG
066A   09                DAD     B          ;IF THERE IS NO CARRY
066B   DA7206            JC      RETN       ;THEN DE < CLOW.
066E   3EFF      ERR:    MVI     A,OFFH     ;SET AND DISPLAY THE
0670   D300              OUT     0          ;ERROR CONDITION.
0672   D1        RETN:   POP     D
0673   C9                RET
                 ;*********************************************
                 ;*                                           *
                 ;*   PROGRAM:  INTERPOLATION                 *
                 ;*                                           *
                 ;*********************************************
0674   21A106    INTER:  LXI     H,DATA
0677   D5        AGAIN:  PUSH    D
0678   4E                MOV     C,M        ;SEARCH MEMORY UNTIL
0679   2C                INR     L          ;THE PROPER INTERVAL IS
067A   46                MOV     B,M        ;FOUND FOR D
067B   EB                XCHG
067C   09                DAD     B
067D   EB                XCHG
067E   DA8906            JC      COM
0681   7D                MOV     A,L
0682   C603              ADI     3
0684   6F                MOV     L,A
0685   D1                POP     D
0686   C37706            JMP     AGAIN
0689   2C        COM:    INR     L
068A   4E                MOV     C,M        ;WHEN THE INTERVAL IS
068B   06FF              MVI     B,OFFH     ;FOUND, MOVE THE COUNT
068D   97                SUB     A          ;DIFFERENCE TO C
068E   EB                XCHG
068F   3C        DIV:    INR     A          ;NOW DIVIDE BY SUCCESSIVE
0690   09                DAD     B          ;SUBTRACTION, WHILE
0691   DA8F06            JC      DIV        ;INCREMENTING A
0694   EB                XCHG
0695   3D                DCR     A
0696   47                MOV     B,A
0697   2C                INR     L          ;SUBTRACT THE RESULT FROM
0698   7E                MOV     A,M        ;THE TEMPERATURE AT THE
0699   90                SUB     B          ;END OF THE INTERVAL, AND
069A   D1                POP     D          ;RETURN WITH THE
069B   57                MOV     D,A        ;TEMPERATURE IN A.
069C   C9                RET
069D   FOEC      CHIGH:  DW      -4880      ;DATA FOR HIGH VALUE
069F   9DFF      CLOW:   DW      -99        ;DATA FOR LOW VALUE
06A1   55F4      DATA:   DW      -2987      ;DATA FOR INTERPOLATION
06A3   43                DB      -189
06A4   0A                DB      10
06A5   97F8              DW      -1897
06A7   93                DB      -109
06A8   14                DB      20
06A9   41FB              DW      -1215
06AB   BC                DB      -68
06AC   1E                DB      30
06AD   DBFC              DW      -805
06AF   D7                DB      -41
06B0   28                DB      40
06B1   DEFD              DW      -546
06B3   E6                DB      -26
06B4   32                DB      50
06B5   8AFE              DW      -374
06B7   EF                DB      -17
06B8   3C                DB      60
06B9   02FF              DW      -254
06BB   F4                DB      -12
06BC   46                DB      70
06BD   4DFF              DW      -179
06BF   F8                DB      -8
06C0   50                DB      80
06C1   7BFF              DW      -133
06C3   FB                DB      -5
06C4   5A                DB      90
06C5   9DFF              DW      -99
06C7   FD                DB      -3
06C8   64                DB      100
                 END
                       SYMBOL TABLE

 * 01

A      0007     AGAIN  0677     B      0000     C      0001
CHECK  0655     CHIGH  069D     CLOW   069F     COM    0689
D      0002     DATA   06A1     DELAY  0648     DIV    068F
E      0003     ERR    066E     H      0004     HEX    0635
HIGH   0624     INTER  0674     L      0005     LOOP   064C
LOW    062D     M      0006     POLL   061E     PSW    0006
RETN   0672     SAVE   0646     SP     0006     ST     0603
START  0639

*STOP* 0
```

PRIME STRIKES AGAIN

After removing most of our data processing work from our Prime computer, we were hoping that it would be able to at least keep up with the Reader Service requests for literature. Several thousand readers suffered through our tussles with the Prime when we tried to have it handle subscriptions . . . a battle we lost, leaving thousands of readers angry in the process.

Now that the Reader Service processing is over two months behind, we admit that the Prime seems to have totally defeated our best intentions. We are moving the data processing involved to an outside agency, so it should get back on schedule in short order. This service is going to be handled by the nationally famous A. C. Nielsen Company.

Subscriptions are now being processed by Fulfillment Associates on Long Island and our complaint department has been dropped from twelve full-time people to one part-timer.

You might think that a magazine which deals with computers would know better than to get into a fix like that, but the sad fact is that it is difficult to know for sure with a new system such as the Prime. We depended mostly on the people from Prime . . . and believed them.

Readers who have requested information in recent months and *not gotten it* should try again . . . Nielsen will get it done, and quickly. Please don't blame the advertisers . . . just us. We are sending the big jobs we had planned for the Prime out to service bureaus such as Nielsen and FAI, and turning most of the other data processing we need over to in-house MSI and TRS-80 systems. We have been finding them easy to use, surprisingly dependable, easy to program, and capable of handling an astounding amount of work.

SOFTWARE FOR OHIO SCIENTIFIC

Over 50 programs for C1, C2, C4, & Superboard, on tape and disk. All come with listings and complete documentation.

GAMES-4K-Tape
CHESS FOR OSI-specify system $19.95
STARFIGHTER 5.95
Real time space war.
SEAWOLFE 5.95
Floating mines, three target ships, etc.
LUNAR LANDER 5.95
With full graphics.
ALIEN INVADERS 6.95
Rows of menacing, munching monsters march on earth.
8K GAMES
BACKGAMMON $9.95
BLACKJACK 6.95
Plays *all* Vegas rules.
Add $1.00 each for Color/Sound.

UTILITIES
C1P CURSOR CONTROL $9.95
Gives real backspace, one key screen clear, *and midline editing.*
RENUMBERER 5.95
SUPERUTILITY 12.95
Has renumberer, Variable table maker and Search.

BUSINESS
SMALL BUSINESS ANALYSIS 15.95
Does profit and loss, quick ratio, breakeven analysis and more. 13 pages of documentation.
STOCK PORTFOLIO 6.95
Keeps track of your investments.

Our $1.00 catalog has free game and utility listings, programming hints and a lot of PEEKs and POKEs and other stuff that OSI forgot to mention—and a *lot* more programs for sale.

DISKS 5" COLOR/SOUND $29.95
DISK 1. STARFIGHTER, ROBOTANK, SEA WOLFE, BOMBER, TEN TANK BLITZ,
DISK 2 BREAKTHROUGH, LUNAR LANDER, ALIEN INVADER, KILLERROBOTS, SLASHBALL.

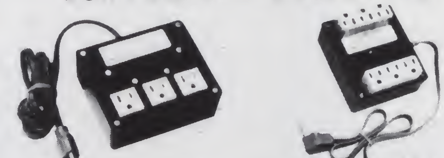
VISA

AARDVARK TECHNICAL SERVICES ✓91

master charge

1690 BOLTON, WALLED LAKE MI 48088 313-624-6316

DISK DRIVE WOES? PRINTER INTERACTION?
MEMORY LOSS? ERRATIC OPERATION?
DON'T BLAME THE SOFTWARE!



ISO-1 ISO-2

Power Line Spikes, Surges & Hash could be the culprit!
Floppies, printers, memory & processor often interact!
Our unique ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash.

*ISOLATOR (ISO-1A) 3 filter isolated 3-prong sockets; integral Surge/Spike Suppression; 1875 W Maximum load, 1 KW load any socket $54.95
*ISOLATOR (ISO-2) 2 filter isolated 3-prong socket banks; (6 sockets total); integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank $54.95
*SUPER ISOLATOR (ISO-3), similar to ISO-1A except double filtering & Suppression $79.95
*ISOLATOR (ISO-4), similar to ISO-1A except unit has 6 individually filtered sockets $93.95
*ISOLATOR (ISO-5), similar to ISO-2 except unit has 3 socket banks, 9 sockets total . . . $76.95
*CIRCUIT BREAKER, any model (add-CB) Add $ 6.00
*CKT BRKR/SWITCH/PILOT any model (-CBS) . . . . Add $11.00

PHONE ORDERS 1-617-655-1532

VISA

ESP Electronic Specialists, Inc. ✓93
171 South Main Street, Natick, Mass. 01760
Dept. KB

• PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET

PET PRODUCTS

Programs — Workbooks for Floppy Disk — for Cassette

See your Dealer

Dealer Inquiries Invited

Put your PET to work!

T I S

PROGRAMS
SW-1* **MAILS** mailing list system
SW-2* **CHECKBOOK** record
SW-3* **ACCOUNTS** keep track of who owes you how much
SW-4 **MEDIT** create and maintain data files
SW-5* **CALENDAR** appointments, meetings at a glance

WORKBOOKS
WB-1 **Getting Started with Your PET** $3.95
WB-2 **PET String and Array Handling** $3.95
WB-3 **PET Graphics** $4.95
WB-4 **PET Cassette I/O** $4.95
WB-5 **Miscellaneous PET Features** $3.95
WB-6 **PET Control and Logic** $3.95

*These programs are special purpose data base management systems. They all can:
- Sort numeric or string fields
- Select based on numeric or string (=)
- Select based on substring match
- Select based on range of entry number

Prices: $9.90 each for programs using cassette storage for data
$12.95 each using sequential floppy disk storage for data.
Price includes 40 - 60 page instruction manual

Add $2.00 for shipping and handling

Money back guarantee

**TIS
P.O. Box 921, Dept. KB
Los Alamos, NM 87544**

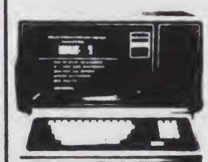
VISA

On bank card orders, give all numbers ✓95

PET is a trademark of Commodore Business Machines

• PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET • PET

Radio Shack® DEALER
COMPUTER SPECIALISTS



15% Discount on TRS-80's - I AND ACCESSORIES

POPULAR 16K LEVEL 2 SYSTEM..........$722.00
FAST 100 cps Centronics 730 PRINTER.... $800.00
HIGHLY RELIABLE LOBO DRIVES......$375.00

MICRO MANAGEMENT SYSTEMS
DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728 ✓100
912-377-7120
"TRS-80 is a Registered Trademark of Tandy Corp."

Theodore C. Hines
Jerry Russell
Rosann Collins
University of NC, Greensboro
Greensboro NC 27412

Quiz Whiz

Teachers! Try this highly adaptable testing program.

A generalized test and quiz program is a handy utility for both education and fun. This one is simple, flexible and readily adaptable to many purposes. We use it to enable students to drill and test themselves on course content at any time during the course. By using a hard-copy terminal, we can also use it to let students take their final examinations at their own convenience, turning in the results only when they pass. With a large pool of questions, attempting to cheat really just results in learning the material.

The Program

The program is written in Processor Technology Extended Cassette BASIC using CRT display but may be easily adapted to other versions of BASIC and to hard-copy terminals.

Questions are entered in relatively free form as DATA statements at the end of the program proper, making it easy to APPEND additional or substitute questions. Questions may be any length, as long as they do not overflow the CRT display. Questions may be various types—multiple choice, true/false or completion questions may be used, and types of questions may be mixed in any order. The program will accept any number of questions.

The number of questions to be presented in each round may be set by changing variable E. Questions are selected randomly from the pool of questions not already used in the current session—that is, the selection routine is so arranged that all questions are presented before any question is repeated. The program offers the users two tries at each question and keeps score of questions answered correctly on first and second tries, reporting the score at the end of each round.

In entering the questions, the programmer finds that each question begins a new DATA statement and is preceded by a "Q," a colon and a space. The question is then entered line for line as you want it to appear on the screen. The answer immediately follows, cannot be longer than one line and is preceded by an "A," a colon and a space. Answers must be all uppercase, and only character-for-character matches are scored as correct. No provision is made for alternative answers.

While our effort was to make the program short and simple, we could not resist providing for varying the computer's response to avoid the kind of deadly sameness that you find so often in programs of this type.

Although the program is copyrighted, permission is hereby granted to copy for personal or nonprofit educational use. Please credit *Microcomputing*. We will be glad to supply cassette tapes in Processor Technology CUTS format at $5 each to cover costs. ■

Dr. Hines is a professor at the University of North Carolina at Greensboro, Library Science/Educational Technology Division of the School of Education. Jerry Russell and Rosann Collins are research associates with the Children's Media Data Bank project at the University.

Program listing.

```
10 REM- TEST PROGRAM
20 CLEAR
30 LET E=5
40 REM- E IS THE NUMBER OF QUESTIONS TO BE PRESENTED
50 REM- IN EACH ROUND
60 DIM A$(65),B$(65),C$(65)
70 REM- COUNT THE NUMBER OF QUESTIONS IN THE POOL
80 READ A$
90 IF A$="END" THEN 120
100 IF A$(1,3)<>"Q: " THEN 80
110 LET N=N+1: GOTO 80
120 RESTORE
130 REM- ARRAY TO KEEP TRACK OF QUESTIONS USED
140 REM- IN EACH SESSION
150 DIM A(N)
160 PRINT "&K"
170 LET A$="TEST": GOSUB 820
180 PRINT : PRINT
190 LET A$="Copyright 1978 by Dr. Theodore C. Hines": GOSUB 820
200 PRINT
210 LET A$="Library Science/Educational Technology Division"
220 GOSUB 820
230 LET A$="School of Education": GOSUB 820
240 LET A$="University of North Carolina at Greensboro"
250 GOSUB 820
260 LET A$="Greensboro, North Carolina 27412": GOSUB 820
270 LET A$="(919) 379-5710": GOSUB 820
280 PAUSE 60
290 PRINT "&K"
300 LET A$="SELF-TEST": GOSUB 820
310 PRINT : PRINT
320 PRINT "     This program presents questions for students"
330 PRINT "to test themselves on their comprehension of"
340 PRINT "course content.  Answer each question appro-"
350 PRINT "priately and then press the RETURN key.  Press"
360 PRINT "RETURN again when you are ready for the next"
370 PRINT "question."
380 PRINT
390 PRINT "     If you are ready for your first question,"
400 PRINT "press RETURN now to get started."
410 INPUT "",X$
420 PRINT "&K"
430 IF E=C THEN 710
440 LET X=INT(RND(0)*N)+1
450 REM- Q3 IS # OF QUESTIONS USED SO FAR IN SESSION
460 IF Q3=N THEN GOSUB 890
470 IF A(X)<>0 THEN 440
480 LET A(X)=1
490 LET Y=0
```

```
500 REM- Y IS THE QUESTION COUNTER IN DATA STATEMENTS
510 READ A$
520 IF A$(1,3)<>"Q: " THEN 510
530 LET Y=Y+1
540 IF Y<>X THEN 510
550 LET Q3=Q3+1: LET C=C+1
560 LET A$=A$(4)
570 PRINT C;".  ";A$
580 READ A$
590 IF A$(1,3)="A: " THEN 610
600 PRINT A$: GOTO 580
610 LET A$=A$(4)
620 PRINT
630 INPUT B$
640 LET M=INT(RND(0)*5)+1
650 IF B$<>A$ THEN 680
660 GOSUB 950
670 GOTO 690
680 GOSUB 1030
690 RESTORE
700 GOTO 410
710 PRINT "&K"
720 PRINT "You got ";R;" right out of ";E;" questions."
730 IF W1=0 THEN 760
740 PRINT "But ";W1;" took two tries - not so good in the"
750 PRINT "case of true/false questions."
760 INPUT "Do you want more questions? (YES or NO) ",A$
770 LET C=0: LET R=0: LET W1=0
780 IF A$="" THEN 760
790 IF A$(1,1)="Y" THEN 290 ELSE 800
800 PRINT "Until the next time, then."
810 END
820 REM-   SUBROUTINE TO CENTER AND PRINT
830 LET C$="                                            "
840 LET W=50-LEN(A$)
850 LET W=INT(W/2)
860 LET C$=C$(1,W)
870 PRINT C$;A$
880 RETURN
890 REM- SUBROUTINE TO CLEAR RECORD OF QUESTIONS AND SCORES
900 LET R=0: LET W1=0: LET Q3=0
910 FOR I2=1 TO N
920   LET A(I2)=0
930 NEXT I2
940 RETURN
950 REM- SUBROUTINE TO CORRECT ANSWERS
960 LET R=R+1
970 ON M GOTO 980,990,1000,1010,1020
980 PRINT "Correct.": RETURN
990 PRINT "That's it!": RETURN
1000 PRINT "Yes, indeed!": RETURN
1010 PRINT "Right you are!": RETURN
1020 PRINT "Very good!": RETURN
1030 REM-SUBROUTINE FOR WRONG ANSWERS
1040 ON M GOTO 1050,1060,1070,1080,1090
1050 PRINT "Sorry.  Try again.": GOTO 1100
1060 PRINT "No.  Another try?": GOTO 1100
1070 PRINT "Oops.  Try over.": GOTO 1100
1080 PRINT "Not right.  Another answer?": GOTO 1100
1090 PRINT "No - give it another try."
1100 INPUT "",B$
1110 IF B$<>A$ THEN 1140
1120 REM- W1 COUNTS # OF ANSWERS CORRECT ON SECOND TRY
1130 LET W1=W1+1: GOSUB 950: RETURN
1140 ON M GOTO 1150,1160,1170,1180,1190
1150 PRINT "No, the correct answer is": GOTO 1200
1160 PRINT "No, the answer you should have given was": GOTO 1200
1170 PRINT "Too bad.  The answer is ": GOTO 1200
1180 PRINT "Still not it.  The answer is": GOTO 1200
1190 PRINT "No, the right answer is"
1200 PRINT A$: RETURN
1210 DATA "Q: If A=B and B=C, does A=C ?(Yes or No)"
1220 DATA "A: YES"
1230 DATA "Q: Job descriptions for primary reading"
1240 DATA "program aides are available from the"
1250 DATA "State Dept. of Public Instruction."
1260 DATA "(True or False)"
1270 DATA "A: FALSE"
1280 DATA "Q: ESLC includes audio-visual materials."
1290 DATA "(True or False)"
1300 DATA "A: TRUE"
1310 DATA "Q: The Fry Readability Formula is appropriate for:"
1320 DATA "     a) Grades 1-3"
1330 DATA "     b) Any level"
1340 DATA "     c) Grades 3-6"
1350 DATA "     d) None of the above"
1360 DATA "(Type a,b,c or d)"
1370 DATA "A: B"
1380 DATA "Q: In North Carolina, paraprofessionals in "
1390 DATA "the schools are unionized. (True or False)"
1400 DATA "A: FALSE"
1410 DATA "Q: Which of the following is not a selection tool?"
1420 DATA "a) Children's Catalog"
1430 DATA "b) the Booklist"
1440 DATA "c) Books in Print"
1450 DATA "(Type a,b,c or d)"
1460 DATA "A: C"
1470 DATA "Q: Roy Roger's horse was named:(------)."
1480 DATA "a) George"
1490 DATA "b) Fred"
1500 DATA "c) Trigger"
1510 DATA "d) Buttercup"
1520 DATA "(Type a,b,c or d)"
1530 DATA "A: C"
1540 DATA "Q: Which of the following was not a U.S. president?"
1550 DATA "a) Herbert Hoover"
1560 DATA "b) Lyndon Johnson"
1570 DATA "c) Rutherford Hayes"
1580 DATA "d) Theodore C. Hines"
1590 DATA "(Type a,b,c or d)"
1600 DATA "A: D"
1610 DATA "Q: In what year did the Battle of Hastings occur?"
1620 DATA "a) 1543"
1630 DATA "b) 1212"
1640 DATA "c) 1066"
1650 DATA "d) 1492"
1660 DATA "(Type a,b,c or d)"
1670 DATA "A: C"
1680 DATA "Q: How many in a baker's dozen ?"
1690 DATA "(Type the number)"
1700 DATA "A: 13"
1710 DATA "Q: What do we call the process by which plants"
1720 DATA "obtain their food?"
1730 DATA "a) protoplasm"
1740 DATA "b) predicament"
1750 DATA "c) photosynthesis"
1760 DATA "d) practice"
1770 DATA "(Type a,b,c or d)"
1780 DATA "A: C"
1790 DATA "Q: Caesar was warned:'Beware the (----) of March."
1800 DATA "(Type the word that goes in the blank.)"
1810 DATA "A: IDES"
1820 DATA "Q: The opposite of positive is: (---------)."
1830 DATA "(Type the appropriate word.)"
1840 DATA "A: NEGATIVE"
1850 DATA "END"
```

Reader Service—see page 257

The Mighty Five-Way Power Supply

Small projects and single board computers will thrive on this easy-to-build power source.

Don Walters
2849 Verle Street
Ann Arbor MI 48104

Necessity is the mother of invention, or so the saying goes. In my case it was the driving force that prompted me to turn an idea for a power-supply design I have had floating around in my mind and make it into a reality. My necessity was a power supply that could be used to power a single-board microcomputer system, such as MOS Technology's KIM-1 or Synertek's VIM-1, or the electronics of a cassette drive system . . . and probably many other similar types of applications in the future.

With the need defined and an idea for a power-supply design, I decided on the particulars of the power supply: (1) what the voltages and current capabilities were going to be (common voltages required for microprocessor and related components, i.e., ± 12, − 9, ± 5 volts), (2) that the design would be simple and flexible (voltages available dependent only on whether their components were installed), how the supply was going to be built (everything, including the power transformer, on a printed circuit board) and (3) what components I wanted to use (easily obtainable, while allowing a low parts count, and mostly locally available).

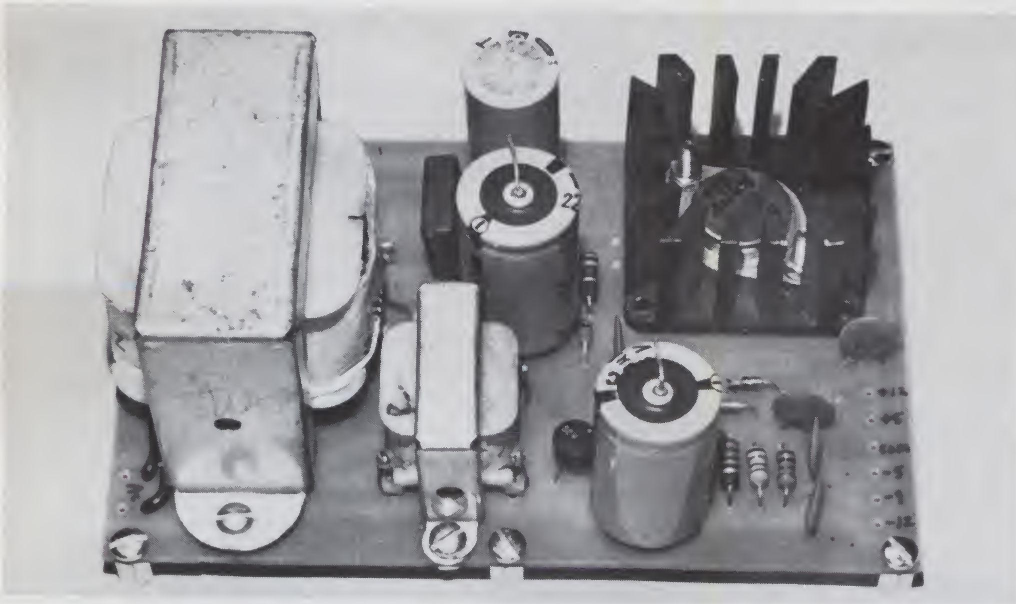


Photo 1. Completed prototype of the power-supply design, which uses zener diodes for the other voltages on the card.

The Results

Photo 1 shows the results of my efforts. The power-supply design is simple, as can be seen by looking at the schematic for it in Fig. 1, and almost all the parts were available locally, except the LM-323K-5 three-terminal-regulator integrated circuit, which was purchased from Jameco Electronics (1021 Howard Ave., San Carlos CA 94070). The design is flexible, in that only the components needed for a desired voltage are required, and the components for unwanted voltages can be left out without affecting the power supply's operation. Also, as can be seen in the photograph, I managed to get all the components, including the power transformer, onto a 4 by 6 inch printed circuit board.

The Design

Because I wanted to keep the design simple and flexible, I decided to make some compromises in both the circuit design and component selection: to use two transformers, to use a 3 Ampere, 5 volt, three-terminal-

Output Voltage	Zener Diode	Resistor Value
5 @ 20 mA	1N5231A	100 Ohms @ ½ Watt
9 @ 40 mA	1N4739A	220 Ohms @ ½ Watt*
12 @ 60 mA	1N4742A	330 Ohms @ ½ Watt
15 @ 60 mA	1N4744A	440 Ohms @ ½ Watt*

*Estimated values based on 5 and 12 volt values. Check zener diode no-load current through it to ensure that is does not exceed zener diode power-handling capability.

Table 1. Current-limiting-resistor values for selected zener diodes, voltages and currents.

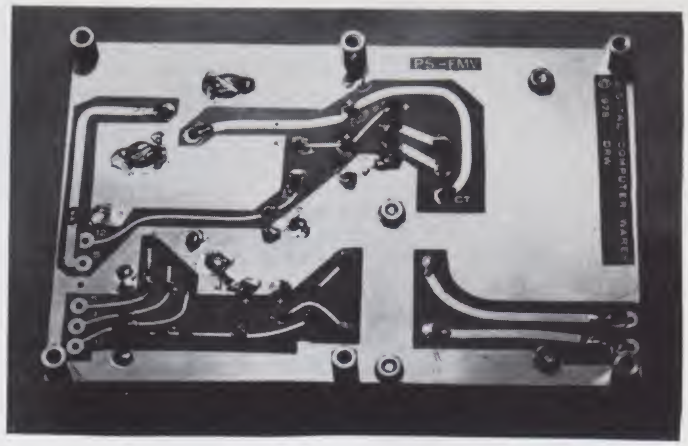


Photo 2. Foil-side view of the prototype power supply printed circuit board. This board is an earlier version and does not have the additional foil pads to allow for either zener diodes or three-terminal regulators.

regulator integrated circuit and to use zener diodes for the remaining voltages (±12, −9, −5 volts). The compromises were really commonsense approaches to reaching some or all of the design goals for the power supply.

I decided to use two transformers instead of one when I found that a source for a transformer with two 12.6 volt secondary windings—one center tapped, the other not, capable of carrying 3 Amperes and 300 milliamperes respectively—was not available. It turns out that the transformer compromise is really a potential advantage because if you only need 5 volts at 2.5 Amperes, you don't have to pay for the extra, unnecessary 12.6 volt, 300 milliampere secondary winding on the transformer. If you do need it eventually, all you have to do is buy the extra transformer. While this capability is not production-quantity economy, it does provide you with options from which you can pick and choose.

Additionally, if you need only 5 volts at 1 Ampere, you can use a power transformer with similar current capability and use a LM-309K 5 volt, 1 Ampere, three-terminal-regulator integrated circuit. Another advantage of using two transformers (in this case) is that they are probably available to you locally at Radio Shack[1] stores and through some of the mail-order firms that advertise in *Kilobaud Microcomputing* and/or *73 Magazine*.

I used a 5 volt, 3 Ampere, three-terminal positive-regulator integrated circuit for the 5 volt, 2.5 Ampere portion of the power supply because it would reduce the number of components needed versus the traditional series pass transistor voltage regulator circuit. Furthermore it would still be capable of supplying 2.5 Amperes of current, with current limiting (an ability to limit the current through the device to a safe value it can handle without damage to it) and thermal shutdown (the device shuts itself off if its operating temperature should go too high). Also, if you only need 5 volts at 1 Ampere you can use the LM-309K regulator without having to redesign the circuit since these two devices are similar and both come in a metal T0-3 style case.

For the remaining voltages the power supply is designed to deliver, zener diodes were initially chosen because their ability to regulate a voltage is adequate for most applications, they are manufactured in a wide variety of voltage ranges (3, 4, 5, 6, 7, 8, 9, 10, 11, 12, . . .), they can handle a moderate amount of current (about 40 milliamperes or so using 1 Watt zener diodes) and they are generally less expensive then three-terminal regulators.

The wide choice of zener diode voltages available allows the power-supply design to deliver whatever voltage your application may require. It should be pointed out, though, that your choices are limited to a maximum of about 17 volts dc because the dc input voltage to the zener diode is about 17.6

T1	12.6 V ac CT @ 3 A transformer
T2	12.6 V ac @ 300 mA transformer
D1	50 V PIV 4 A diode bridge
D2	50 V PIV 1 A diode bridge
D3, D4, D5, D6	see Table 1 for zener diode part numbers
C1	2000 uF 35 V electrolytic capacitor
C3	2000 uF 35 V electrolytic capacitor
C2	1000 uF 35 V electrolytic capacitor
C4	33 uF solid tantalum capacitor 16 V
C5	.1 uF disk ceramic capacitor 50 V
C6	.1 uF disk ceramic capacitor
C7	.1 uF disk ceramic capacitor
C8	.1 uF disk ceramic capacitor
R1, R2, R3, R4	see Table 1 for resistor values
IC1	National Semiconductor LM-323K-5 5 V 3 A 3-terminal-regulator integrated circuit
H1	TO-3 style heat sink. Thermalloy 6004B-2 or equivalent (large surface area heat sink needed for the 3 A regulator).

Parts list for Fig. 1.

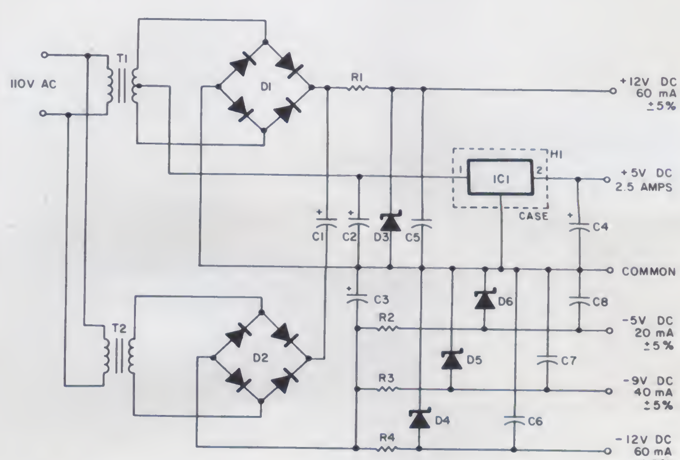


Fig. 1. Schematic and parts list of the power supply using zener diodes.

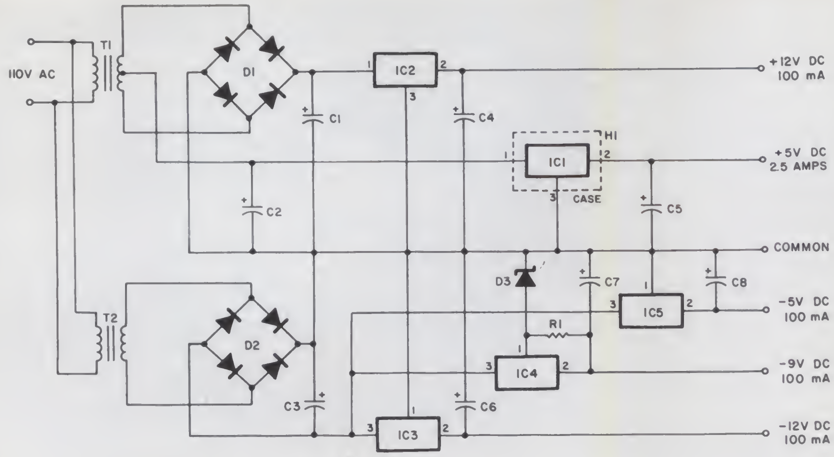


Fig. 2. Schematic and parts list of the power supply using three-terminal regulators.

T1	12.6 V ac CT @ 3 A transformer
T2	12.6 V ac @ 400 mA transformer
D1	50 V PIV 4 A diode bridge
D2	50 V PIV 1 A diode bridge
D3	3.9 V 1 W zener diode 1N4730A
IC1	National Semiconductor LM-323K-5 5 V 3 A 3-terminal positive-regulator integrated circuit
IC2	LM78L12CH 12 V 100 mA 3-terminal positive-regulator integrated circuit
IC3	LM320H-12 12 V 200 mA 3-terminal negative-regulator integrated circuit
IC4	LM320H-5.2 5.2 V 200 mA 3-terminal negative-regulator integrated circuit
IC5	LM320H-5 5 V 200 mA 3-terminal negative-regulator integrated circuit
R1	330 Ohm ½ W carbon resistor
C1	1000 uF 35 V electrolytic capacitor
C2	1000 uF 35 V electrolytic capacitor
C3	1000 uF 35 V electrolytic capacitor
C4	33 uF 16 V solid tantalum capacitor
C5	33 uF 16 V solid tantalum capacitor
C6	33 uF 16 V solid tantalum capacitor
C7	33 uF 16 V solid tantalum capacitor
C8	33 uF 16 V solid tantalum capacitor
H1	TO-3 style heat sink. Thermalloy 6004B-2 or equivalent (large surface area heat sink needed for the 3 ampere regulator).

Parts list for Fig. 2.

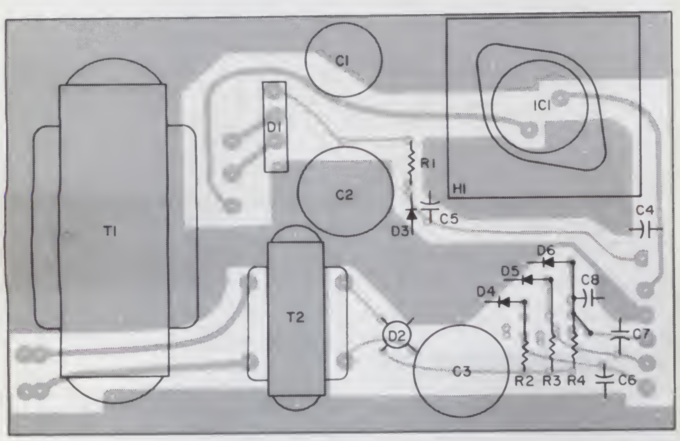


Fig. 3. Component placement on the component side of the PC board for the supply using zener diodes.

volts dc.

To get a particular regulated voltage from the power supply, all you have to do is select the appropriate zener diode for the voltage and current you need and then calculate the value of the current-limiting resistor needed. Table 1 lists the values of current-limiting resistors for a specific zener diode for the common voltages of 15, 12, 9 and 5 volts, which are needed for operational amplifiers, microprocessors and their related components.

As an afterthought I decided that in some applications a higher current capability of about 100 milliamperes or more might be necessary for the ±12, −9, −5 volt outputs of the power supply. So I modified the design of the power supply in Fig. 1 to allow for three-terminal positive- and negative-regulator integrated circuits to be used in place of the zener diodes (see Fig. 2) when the intended application required the higher current capability.

This change rippled a bit in that the power-supply printed circuit board artwork had to be changed to allow for either zener diodes or three-terminal regulators to be used on the printed circuit board. But such ripples are bound to occur when you rush to turn an idea into a reality.

It Looks Good So Far, But How Does the Circuit Work?

For the description of how the circuit works, refer to Fig. 1. Transformer T1's full secondary winding (12.6 volts ac) is fed to a full wave rectifier module, D1. The rectified output, which is now approximately 17.6 volts dc but with a 120 Hz ripple, is fed to an electrolytic filter capacitor, C1, which filters out the 120 Hz ripple. The filtered dc voltage is then regulated (held to a fixed voltage) by zener diode D3, while R1 limits the no-load current through the zener diode to a safe value.

Since zener diodes are nonlinear devices they generate some noise. Although the noise is just visible on an oscilloscope, it could possibly cause problems in the circuitry powered by the zener supply. So disk ceramic capacitor C5 is used to filter out this noise.

The center tap of transformer T1's secondary winding in conjunction with diode bridge D1 supplies about 8.8 volts dc with a 120 Hz ripple, which is half the voltage value of T1's full rectified secondary winding voltage. This voltage is filtered of the 120 Hz ripple by electrolytic filter capacitor C2. The filtered dc voltage is then input to IC1, a 5 volt, 3 Ampere, three-terminal, positive-voltage-regulator integrated circuit. The use of T1's secondary tap is desirable because the 8.8 volts dc available will mean a lower voltage drop across IC1 as compared to the drop if the input to the IC were 17.6 volts.

This means that the IC should run a tad cooler because there is a smaller voltage drop across the IC. Capacitor C5 is used to filter out any noise generated by the fast-switching logic in the IC.

The circuitry composed of T2, D2 and C3 functions the same as that of T1, D1 and C1, except that the output of C3 is negative with respect to common since this section of the power supply will be supplying the needed negative voltages. Current-limiting resistors R2, R3 and R4 serve the same function as R1 described above.

Zener diodes D4, D5 and D6 also serve the same function as D3, except they regulate the dc voltage input to them to different voltage values according to their regulated voltage rat-

ings. Disk ceramic capacitors C6, C7 and C8 also serve the same function as C4: to filter out any noise generated by the zener diodes.

When three-terminal-regulator integrated circuits are used for ±12, −9 and −5 volt outputs rather than zener diodes, the explanation of how the circuit works is not that much more complicated. Refer to the schematic of Fig. 2. All the circuit components up to the input of the three-terminal regulators function the same as they did in the schematic of Fig. 1.

The three-terminal regulator ICs (IC2, IC3, IC4 and IC5) function essentially the same as IC1, except that their output current capability is about 100 milliamperes without additional heat sinking, and IC3, IC4 and IC5 regulate a negative voltage. IC4, however, is slightly different from the other voltage regulators because it does not regulate the voltage input to it directly.

Zener diode D3 and resistor R1 are used to change IC4's ground reference point by about −3.9 volts dc. When combined with IC4's −5.2 volt regulation, this voltage change causes the output voltage from IC4 to be −9.1 volts rather than −5.2 volts. Capacitors C6, C7 and C8 are solid tantalum capacitors and are required by the negative regulators IC3, IC4 and IC5 to stabilize their operation and filter out any noise generated by the fast-switching logic in the ICs.

It Looks Good; How Can I Build One?

The physical size and choice of components used allow much flexibility in putting the power supply together—especially when you realize that you probably don't need all the voltages that the power-supply design can be set up to deliver.

My personal preference was to build the power supply, including the power transformers, on a 4 by 6 inch printed circuit board[2]. However, you could just as easily build the power supply on a reasonable size piece of perfboard. Fig. 3 is a parts-placement diagram for a printed circuit board version of the power supply that uses zener diodes, while Fig. 4 is a parts-placement diagram for a version of the power supply using three-terminal-regulator ICs. Depending on the components used (i.e., zener diodes or three-terminal regulators), either parts-placement diagram could be used as a guide for either printed circuit board or perfboard assembly of the power supply described here. Fig. 5 gives some component-mounting details.

Remember that the LM-323K-5 regulator *needs* to be adequately heat-sinked, preferably with a thin smear of silicone grease spread on its bottom, before being mounted onto the heat sink. Also, if you are building the power supply on a piece of perfboard, remember to use #18 stranded or heavier wire for all 5 volt, 2.5 Ampere runs, including the run from the power supply to whatever it is power-

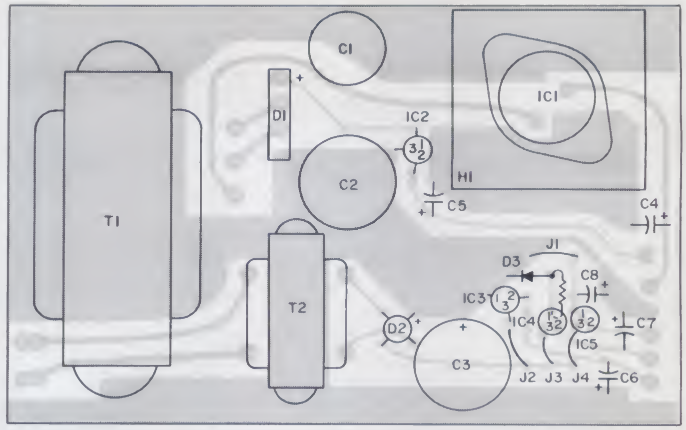


Fig. 4. Component placement on the component side of the PC board for the supply using three-terminal regulators.

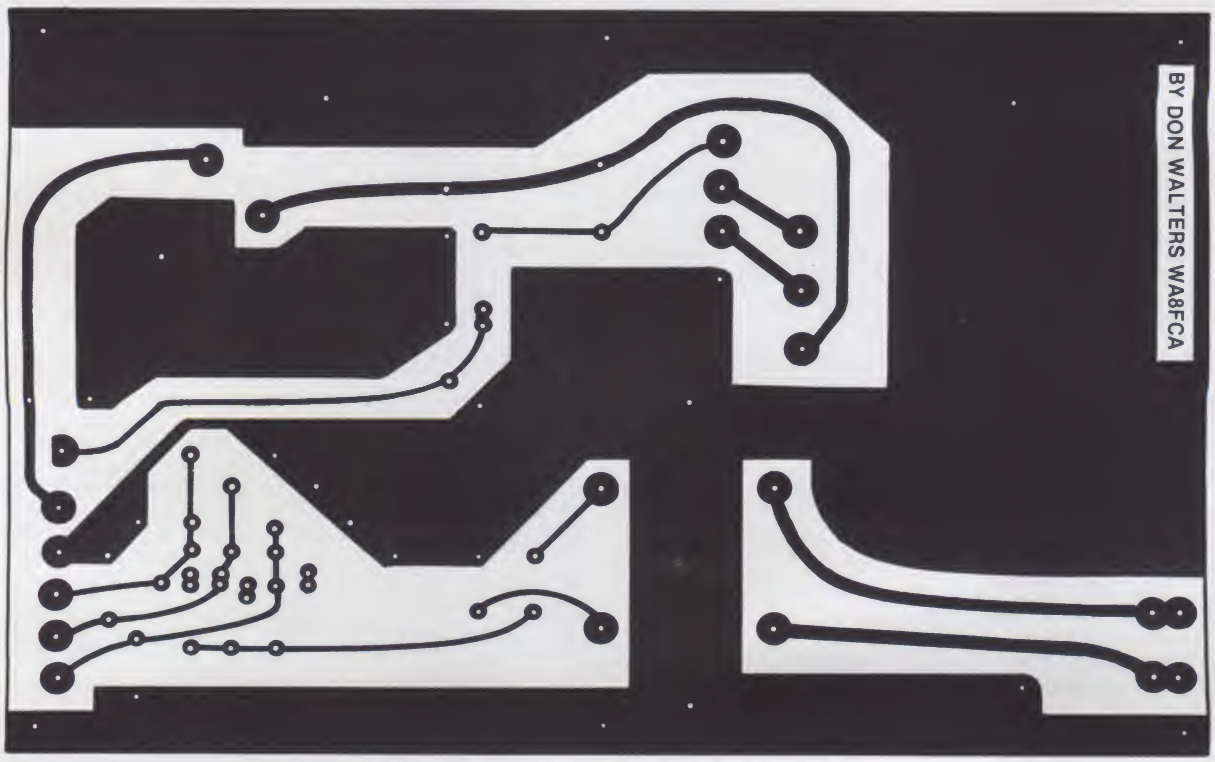


Fig. 6. Power supply printed circuit board foil pattern, foil-side view. The dark areas are foil traces; light areas are etched out portions of the PC board.

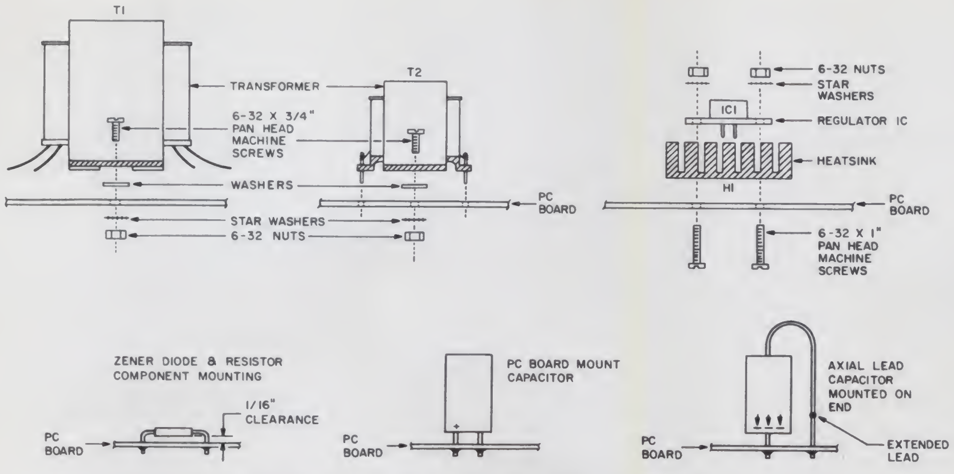


Fig. 5. Power supply component mounting details.

ing. The wire may need to carry up to 2.5 Amperes, so it must be heavy enough to safely handle that current.

Is That All There Is to It?

That's all there is to this power supply, which is what makes the design flexible and easy to build.

The power-supply design, in the form of two prototypes (using zener diodes for ± 12, − 9 and − 5 volts), has been built on printed circuit boards and put into operation powering a friend's VIM-1 single-board microcomputer system and powering the electronics of a cassette drive system under development. In both applications the power supplies have performed well and without problems.

Probably the most attractive feature of this design is the cost. A fully stuffed power supply using zener diodes for ± 12, − 9 and − 5 volts costs about $35 to build, while the same supply using three-terminal regulators for the same voltages will cost about $40 to build. These prices are substantially less than that for a comparable commercial power supply.

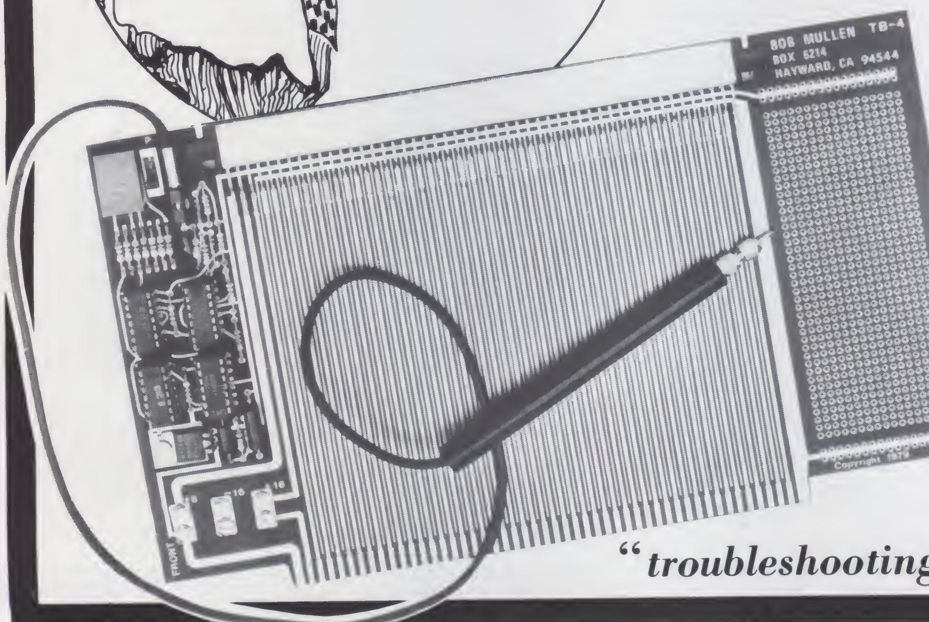
If you need a power supply to run an AIM-1, Cosmac Elf, KIM-1, VIM-1 or other single- or two-board microcomputer system — or for just experimenting with TTL, operational amplifiers or microcomputers and related components — this power-supply design should be of interest to you. ■

References

1. Radio Shack sometimes has a quality-control problem with their transformers. Therefore, it is good practice to check the transformer windings with a VOM for open or shorted windings before leaving the store. This may save you from having to make a trip back to the store to exchange a defective transformer.
2. If sufficient demand exists for the printed circuit board for this power-supply design, the author will have a quantity of the PC boards made, at a reasonable price, for distribution to those interested.

MULLEN Computer Products

"The time has come," the Walrus said, "to talk of many things... of the Mullen TB-4 of the S-100 bus Machine..."



MULLEN TB·4

S-100 EXTENDER/LOGIC PROBE

THIS MULLEN S-100 EXTENDER. has interlaced ground traces which reduce coupling between signal traces, and improves performance with higher clock frequency boards.

THE NEW LOGIC PROBE, uses a brighter, easier to read display.

New proposed IEEE Standard edge connector label, and all the fine quality parts and documentation that have always been part of Mullen Products are included

Kit $49.
Assm/tested $64.

"troubleshooting tool"

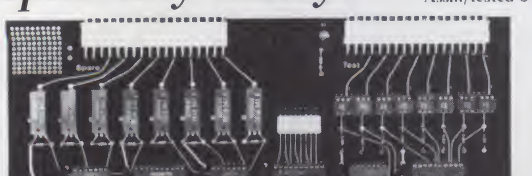
S·100 CONTROLLER BOARD

8 relay — OUTPUTS
8 opto-isolator — INPUTS
256 switch selectable addresses

Our S-100 CONTROLLER is used in laboratories, at universities, and in industry, in hundreds of applications, and maybe the answer to your control problem. Complete programming and operation instructions included.

"exceptionally handy"

Kit $129.
Assm/tested $179.



For higher power applications a 500W AC POWER MODULE is available for $15.

H8 EXTENDER BOARD

Our HTB-O lets H8 owners troubleshoot their boards faster and easier.

Each board can be extended above the computer for complete access to all circuits and components.

HTB·O

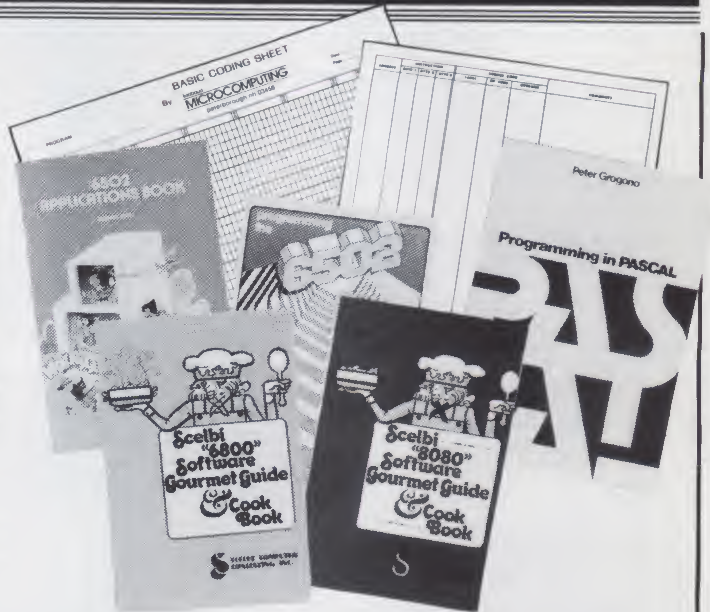
$39 kit only

PLEASE ORDER KITS BY NAME (H8 OR S-100). SEND TO: MULLEN COMPUTER PRODUCTS, BOX 6214, HAYWARD, CA 94544
OR PHONE (415) 783-2866. VISA/MASTERCHARGE ACCEPTED. / INCLUDE $1.50 FOR SHIPPING & HANDLING.
CALIFORNIA RESIDENTS ADD TAX.

37

Order Direct or Contact your Local Computer Store.

kb microcomputing book nook



●**PROGRAMMING IN PASCAL**—BK1140—by Peter Grogono. The computer programming language PASCAL was the first language to embody in a coherent way the concepts of structured programming, which has been defined by Edsger Dijkstra and C.A.R. Hoare. As such, it is a landmark in the development of programming languages. PASCAL was developed by Niklaus Wirth in Zurich; it is derived from the language ALGOL 60 but is more powerful and easier to use. PASCAL is now widely accepted as a useful language that can be efficiently implemented, and as an excellent teaching tool. It does not assume knowledge of any other programming language; it is therefore suitable for an introductory course. $11.50.*

●**8080 SOFTWARE GOURMET GUIDE AND COOKBOOK**—BK1102—If yu have been spending too much time developing simple routines for your 8080, try this new book by Scelbi Computing and Robert Findley. Describes sorting, searching, and many other routines for the 8080 user. $10.95.*

●**6800 SOFTWARE GOURMET GUIDE AND COOKBOOK**—BK1075—Like its culinary cousin, *The 8080 Gourmet Guide*, this new book by Scelbi Computing and Robert Findley describes sorting, searching and other routines—this time for the 6800 user. $10.95.*

●***MICROCOMPUTING* CODING SHEETS** *Microcomputing's* dozen or so programmers wouldn't try to work without these handy scratch pads, which help prevent the little errors that can cost hours and hours of programming time. Available for programming is **Assembly/Machine Language** (PD1001), which has columns for address, instruction (3 bytes), source code (label, op code, operand) and comments; and for **BASIC** (PD1002) which is 72 columns wide. 50 sheets to a pad. $2.39.*

●**PROGRAMMING THE 6502 (Second Edition)**—BK1005—Rodnay Zaks has designed a self-contained text to learn programming, using the 6502. It can be used by a person who has never programmed before, and should be of value to anyone using the 6502. The many exercises will allow you to test yourself and practice the concepts presented. $12.95.*

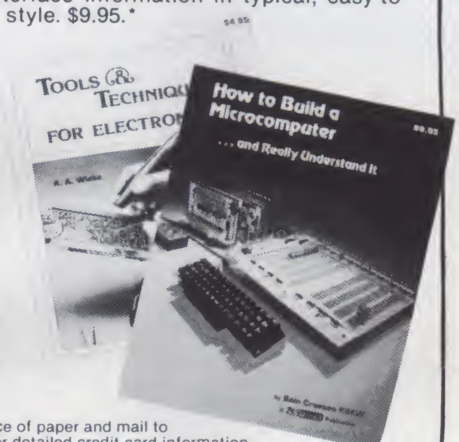
●**6502 APPLICATIONS BOOK**—BK1006—Rodnay Zaks presents practical-application techniques for the 6502 microprocessor, assuming an elementary knowledge of microprocessor programming. You will build and design your own domestic-use systems and peripherals. Self-test exercises included. $12.95.*

●**MICROPROCESSOR INTERFACING TECHNIQUES**—BK1037—by Austin Lesea & Rodnay Zaks will teach you how to interconnect a complete system and interface it to all the usual peripherals. It covers hardware and software skills and techniques, including the use and design of model buses such as the IEEE 488 or S100. $15.95.*

●**TTL COOKBOOK**—BK1063—by Donald Lancaster. Explains what TTL is, how it works, and how to use it. Discusses practical applications, such as a digital counter and display system, events counter, electronic stopwatch, digital voltmeter and a digital tachometer. $9.50.

●**CMOS COOKBOOK**—BK1011—by Don Lancaster. Details the application of CMOS, the low power logic family suitable for most applications presently dominated by TTL. Required reading for every serious digital experimenter! $10.50.*

●**TVT COOKBOOK**—BK1064—by Don Lancaster. Describes the use of a standard television receiver as a microprocessor CRT terminal. Explains and describes character generation, cursor control and interface information in typical, easy-to-understand Lancaster style. $9.95.*



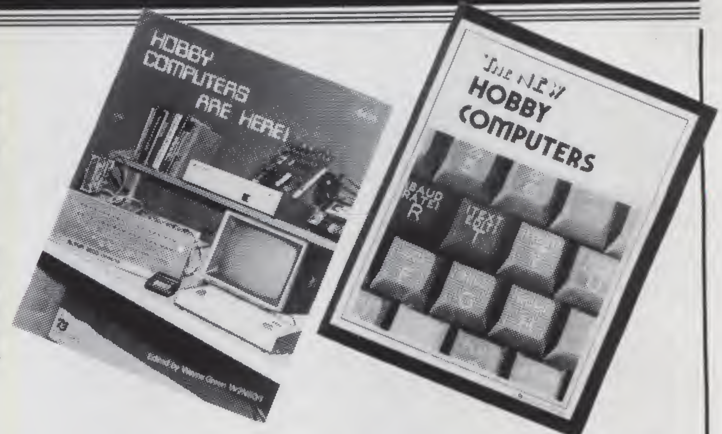
●**HOW TO BUILD A MICROCOMPUTER—AND REALLY UNDERSTAND IT**—BK7325—by Sam Creason. The electronics hobbyist who wants to build his own microcomputer system now has a practical "How-To" guidebook. This book is a combination technical manual and programming guide that takes the hobbyist step-by-step through the design, construction, testing and debugging of a complete microcomputer system. Must reading for anyone desiring a true understanding of small computer systems. $9.95.*

●**TOOLS & TECHNIQUES FOR ELECTRONICS**—BK7348—is an easy-to-understand book written for the beginning kit builder as well as the experienced hobbyist. It has numerous pictures and descriptions of the safe and correct ways to use basic and specialized tools for electronic projects as well as specialized metal working tools and the chemical aids which are used in repair shops. $4.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department ● Peterborough NH 03458. Be sure to include check or detailed credit card information. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

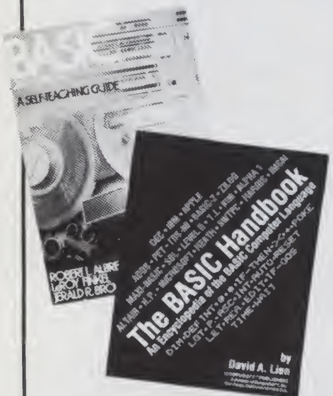
FOR TOLL FREE ORDERING CALL 1-800-258-5473

kb microcomputing book nook



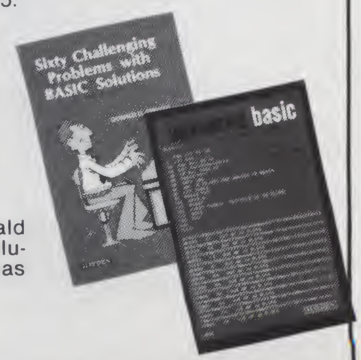
●**HOBBY COMPUTERS ARE HERE!** – BK7322 – If you (or a friend) want to come up to speed on how computers work . . . hardware and software . . . this is an excellent book. It starts with the fundamentals and explains the circuits, and the basics of programming. This book has the highest recommendations as a teaching aid for newcomers. $4.95.*

●**THE NEW HOBBY COMPUTERS** – BK7340 – This book takes it from where "HOBBY COMPUTERS ARE HERE!" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards . . . and much, much more! Don't miss this tremendous value! Only $4.95.*



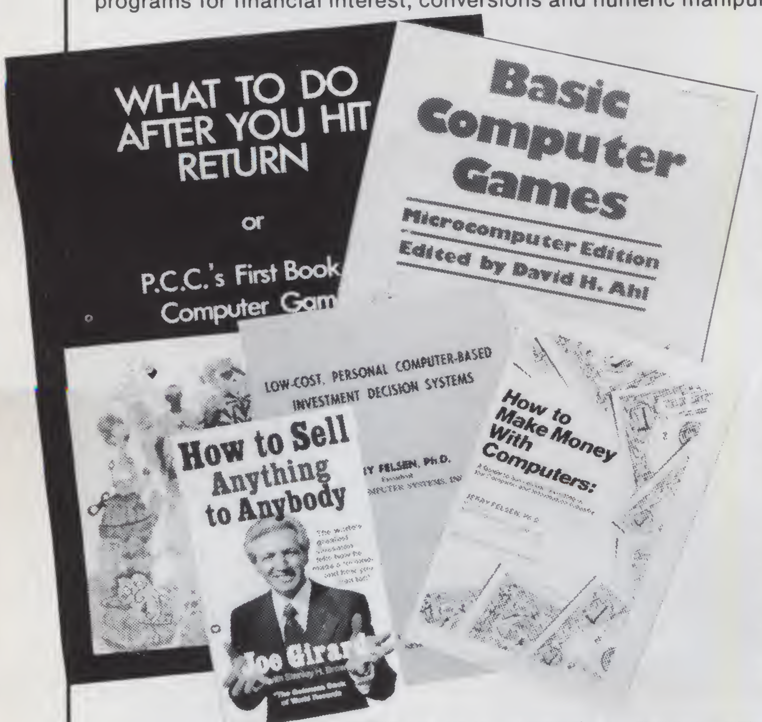
●**THE BASIC HANDBOOK** – BK1174 – by David Lien. This book is unique. It is a virtual ENCYCLOPEDIA of BASIC. While not favoring one computer over another, it explains over 250 BASIC words, how to use them and alternate strategies. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word or combination of words. That's where the HANDBOOK comes in. It helps you get the most from your computer, be it a "bottom-of-the-line" micro or an oversized monster. $14.95.*

●**BASIC NEW 2ND EDITION** – BK1081 – by Bob Albrecht. Self-teaching guide to the computer language you will need to know for use with your microcomputer. This is one of the easiest ways to learn computer programming. $6.95.*



●**ADVANCED BASIC** – BK1000 – Applications and problems by James Coan is for those who want to extend their expertise with BASIC. Offers advanced techniques and applications. $9.65.*

●**SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS (2nd Edition)** – BK1073 – by Donald Spencer, provides the serious student of BASIC programming with interesting problems and solutions. No knowledge of math above algebra required. Includes a number of game programs, as well as programs for financial interest, conversions and numeric manipulations. $6.95.*



●**LOW-COST, PERSONAL COMPUTER-BASED INVESTMENT DECISION SYSTEMS** – BK1101 – Use this guidebook by Man-Computer Systems, Inc.'s president, Jerry Felsen, to develop inexpensive personal computer systems that can help you make better investment decisions. $15.00.*

●**HOW TO SELL ANYTHING TO ANYBODY** – According to *The Guinness Book of World Records*, the author, Joe Girard, is "the world's greatest salesman." This book reveals how he made a fortune – and how you can, too. $2.25.* (BK7306)

●**HOW TO MAKE MONEY WITH COMPUTERS** – BK1003 – In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.*

●**WHAT TO DO AFTER YOU HIT RETURN** – BK1071 – PCC's first book of computer games . . . 48 different computer games you can play in BASIC . . . programs, descriptions, many illustrations. Lunar Landing, Hammurabi, King, Civel 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $10.95.*

●**BASIC COMPUTER GAMES** – BK1074 – Okay, so once you get your computer and are running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department ● Peterborough NH 03458. Be sure to include check or detailed credit card information. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

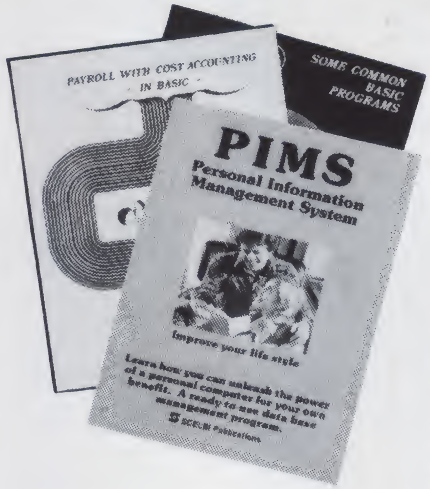
kb microcomputing book nook

● **AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—BK1130—The Beginner's Book—Written for readers who know nothing about computers—for those who have an interest in how to use computers—and for everyone else who must live with computers and should know a little about them. The first in a series of 4 volumes, this book will explain how computers work and what they can do. Computers have become an integral part of life and society. During any given day you are affected by computers, so start learning more about them with Volume 0. $7.95.*

● **VOL. I**—BK1030—Dedicated to the basic concepts of microcomputers and hardware theory. The purpose of Volume I is to give you a thorough understanding of what microcomputers are. From basic concepts (which are covered in detail), Volume I builds the necessary components of a microcomputer system. This book highlights the difference between minicomputers and microcomputers. $8.50.*

● **VOL. II**—BK1031 (loose leaf)—$25.00*; BK1040 (with binder)—$30.00*—Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips.

● **VOL. III**—BK1132 (loose leaf)—$15.00*; BK1133 (with binder)—$20.00.* Contains descriptions of all support devices that can be used with any microprocessor.



TOLL FREE ORDERING
CALL 1-800-258-5473

● **PAYROLL WITH COST ACCOUNTING—IN BASIC**—BK1001— by L. Poole & M. Borchers, includes program listings with remarks, descriptions, discussions of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flowcharts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor of distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941. $20.00.*

● **SOME COMMON BASIC PROGRAMS**—BK1053—published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the stand-alone mode. $12.50 paperback.*

● **PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM**—BK1009—Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $9.95.*

Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department ● Peterborough NH 03458. Be sure to include check or detailed credit card information. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

only a few left...

1977 KILOBAUD: the COMPLETE COLLECTION for $20

That's right, we're liquidating the last issues of *Kilobaud* we have been saving . . . these are COLLECTOR'S COPIES . . . with several of the individual copies worth more than we're asking for the complete year.

This includes the very hard-to-get #1 issue which has been going for as much as $25 at auctions . . . and a library shelf box . . . the whole works, complete. When these rare copies are gone, that will be the end of it. This is your last chance to get a complete collection of Kilobaud . . . starting from the first

During this first year *Kilobaud* had 224 feature articles, including 85 with complete program listings. Just as a source of programs this is a bargain.

The $20 price includes the library shelf box and shipping charges within the USA. You can call our toll free number and order this special: (800) 258-5473. Have your credit card handy.

Please specify #KB1977 when ordering.

DEALER DIRECTORY

Tell your dealer you saw his ad in *Microcomputing*.

Los Angeles CA
Featuring: PolyMorphic, North Star, Imsai, Cromemco, Extensys, Speechlab products and Poly-88 Users Group software exchange. All products 10-20% off list. We won't be undersold! **A-A-A-A Discount Computer How's, 1477 Barrington, Suite 17, Los Angeles CA 90025, 477-8478.**

Mountain View CA
Systems for business, industry and hobbyist. Five terminals, ten printers and five mainframes on display. Superbrain, Horizon Quad, Compucolor II, Equibox and Altos Computer System with hard disk. Much software incl. CP/M, TRS-80 and PET. **Digital Deli Computer Store, 80 W. El Camino Real, Mt. View CA 94040, 961-2670.**

North Hollywood CA
Wholesale prices to dealers & computer club members. Computers, printers, terminals, modems, boards, etc. **Patio Computer Sales Co., 5437 Laurel Canyon Bl., #208, N. Hollywood CA 91607, 762-0020.**

Pompano Beach FL
Business systems, personal systems, whatever the application, we can help. Consulting, programming, education and maintenance. Service, support and professionalism at affordable prices. **Computer Age Inc., 1308 N. Federal Hwy., Pompano Beach FL 33062, 946-4999.**

Venice FL
Discount prices & professional service: Cromemco, North Star, Vector Graphic, DEC, TI, Thinker Toys, Intertube, Soroc, Centronics, NEC, Selectric interfaces, Microdasys. Complete business & medical billing software available. MicroAge & Serendipity software discounted. **Sara-Tech Electronics, Inc., Computer Division, PO Box 692, Venice FL 33595, 485-3559.**

Aurora IL
Microcomputer systems for home or business, peripherals, software, books & magazines. Apple, Hewlett-Packard, North Star, Cromemco systems. IDS-440G printer w/Apple graphics, New HP-85 & HP calculators. **Farnsworth Computer Center, 1891 N. Farnsworth Ave., Aurora IL 60505, 851-3888.**

Chicago IL
Computer Hardware/Software Specialists for home and business. Largest selection of computer books, magazines and copyrighted software in Chicago Metro area. Experienced factory-trained service department. Feature Apple and Alpha Microsystems and accessories. **Data Domain of Schaumburg, 1612 E. Algonquin Road., Schaumburg IL 60195, 397-8700.**

Granite City IL
Come see our full line of Apple II computers & peripherals, North Star and Cromemco business systems. Full line of components, books and magazines. Ten minutes from St. Louis MO. **Computer Station, #12 Crossroads Plaza, Granite City IL 62040, 452-1860.**

Naperville IL
Computer systems design, programming and consultation by computer experts. Dealer for SSM, Integrand, Tarbell, Ithaca Intersystems, Verbatim, Diablo and others. Discount prices on many items. **Wilcox Enterprises, 25W178-39th St., Naperville IL 60540, 420-8601.**

Laurel MD
Apple & Exidy Sorcerer. Authorized dealer for both. We also sell subscriptions to "The Source" timeshare service. Plus full line of ham equipment. **The Comm Center, Laurel Plaza, Rte. 198, Laurel MD 20810, 792-0600.**

Hobbyists and Businessmen These dealers can supply your needs in the home and business computer market. Call the one nearest you!

Worcester MA
Computer products for personal and business systems. Largest selection of software for TRS-80, Apple, PET. Authorized Apple sales and service. **Computer Packages Unlimited, Centerwood Terrace, Route 12, West Boylston MA 01583, 835-3428.**

Worcester MA
Complete business and personal systems on Data General, microNova and Apple. Software, books, magazines, accessories and supplies. Custom business programming available. Authorized Apple service. Downtown. **The Computer Place Inc., 11 Harvard St., Worcester MA 01609, 755-5387.**

Garden City MI
Complete systems for business, professional and personal applications. Custom programming available. Apple II, North Star, Vector Graphic and other lines of microcomputers, software, books, components. **Computer Center, 28251 Ford Rd., Garden City MI 48135, 422-2570.**

St. Louis MO
Experimenters' Paradise. Electronic and mechanical components. Computer People, Audio People, Hams, Robot Builders, Experimenters. Open six days a week. **Gateway Electronics Corp., 8123-25 Page Blvd., St. Louis MO 63130, 427-6116.**

Albuquerque NM
Computer Software for business and home. TRS-80, Apple, North Star. Also Centronics printers. Discounts available. **Legey and Castillo Associates, 2908 Tahiti Ct. N.E., Albuquerque NM 87112, 298-5734.**

New York NY
S-100 New York City systems provided for business and office applications. We also seek experienced S-100 programmers on a part-time basis. **Olcott International & Co., PO Box 3014, Weehawken NJ 07087, 863-4200.**

Rome-Utica NY
Ohio Scientific authorized dealer. Computers for personal, business, education and industry. Software for word processing, education, business, personal and games. **Comp-u-Ware, 9057 Main St., Westernville NY 13486, 827-4487.**

Akron OH
We've got it all. Business systems. Personal systems. Software packages. Custom programming. Terminals. Printers. Service and books. Easy freeway access. 10 AM to 6 PM Monday-Saturday. **The Basic Computer Shop, Fairlawn Plaza, 2671 West Market St., Akron OH 44313, 867-0808.**

Portland OR
Ohio Scientific specialists for business and personal computers. Local service. Terminals, printers, custom programming. Full OSI product line on display! 10AM to 6PM M-F. **Fial Computer, 11266 SE 21st Ave., Milwaukie OR 97222, 654-9574.**

Kingston PA
We support Level II and Model II. Books, magazines, programs, parts, accessories, peripherals, free literature, free seminars, cassettes, floppies, filters, transformers, caps, chips, CRTs. **Artco Electronics, 302 Wyoming Ave., Kingston PA 18704, 287-1014.**

Philadelphia/So. Jersey
North Star, fast delivery possible. Heath terminals, Intertube II, immediate delivery. Free video terminal comparison, Intertec's SuperBrain, all Centronics printers, Omnitec data modems/couplers, NCR terminals. **L & S Distributors, 44 So. Locust, Marlton NJ 08053, 983-7444.**

Dealers: Listings are **$15** per month in prepaid quarterly payments, or one yearly payment of **$150**, also prepaid. Ads include 25 words describing your products and services plus your company name, address and phone. (No area codes or merchandise prices, please.) Call Marcia at 603-924-7138 or write *Kilobaud MICROCOMPUTING*, Ad Department, Peterborough NH 03458

P.O. Box 4430S, Santa Clara, CA 95054
Will calls: 2322 Walsh Ave.
(408) 988-1640

Quest ELECTRONICS

✓44

NEW PRODUCTS!

Super Color S-100 Video Kit $99.95
Expandable to 256 x 192 high resolution color graphics. 6847 with all display modes computer controlled. Memory mapped. 1K RAM expandable to 6K. S-100 bus 1802, 8080, 8085, Z80 etc.

Gremlin Color Video Kit $59.95
32 x 16 alpha/numerics and graphics; up to 8 colors with 6847 chip; 1K RAM at E000. Plugs into Super Elf 44 pin bus. Not expandable to high resolution Graphics.

Elf II Adapter Kit $24.50
Plugs into Elf II providing Super Elf 44 and 50 pin bus plus S-100 bus expansion (With Super Expansion). High and low address displays, state and mode LED's optional **$18.00.**

1802 16K Dynamic RAM Kit $149.00
1802/S-100 expandable to 32K, Hidden refresh w/clocks up to 4 MHz w/no wait states Addl. 16K RAM **$79.00.**

Quest Super Basic

Quest, the leader in inexpensive 1802 systems announces another first. Quest is the first company worldwide to ship a **full size Basic** for 1802 systems. A complete function **Super Basic** by **Ron Cenker** including floating point capability with scientific notation (number range ± .17E38), 32 bit integer ± 2 billion; Multi dim arrays, String arrays; String manipulation; Cassette I/O, Save and load, Basic, Data and machine language programs; and over 75 Statements, Functions and Operators.

Easily adaptable on most 1802 systems. Requires 12K RAM minimum for Basic and user programs. Cassette version in stock now. ROM versions coming soon with exchange privilege allowing some credit for cassette version.

Super Basic on Cassette **$40.00**

Tom Pittman's 1802 Tiny Basic Source listing now available. Find out how Tom Pittman wrote Tiny Basic and how to get the most out of it. Never offered before. **$19.00**

S-100 4-Slot Expansion **$ 9.95**

Super Monitor VI.I Source Listing **$15.00**

Coming Soon: Assembler, Editor, Disassembler, DA/AD, Super Sound/Music, EPROM programmer, Stringy Floppy Disc System.

RCA Cosmac Super Elf Computer $106.95

Compare features before you decide to buy any other computer. There is no other computer on the market today that has all the desirable benefits of the **Super Elf** for so little money. The Super Elf is a small single board computer that does many **big** things. It is an excellent computer for training and for learning programming with its machine language and yet it is easily **expanded with additional memory, Full Basic, ASCII Keyboards, video character generation, etc.**

Before you buy another small computer, see if it includes the following features: ROM monitor; State and Mode displays; Single step; Optional address displays; Power Supply; Audio Amplifier and Speaker; Fully socketed for all IC's; Real cost of in warranty repairs; Full documentation.

The **Super Elf** includes a **ROM** monitor for program loading, editing and execution with **SINGLE STEP for program debugging** which is not included in others at the same price. With **SINGLE STEP** you can see the microprocessor chip operating with the **unique Quest** address and data bus displays **before, during** and **after** executing instructions. Also, CPU mode and instruction cycle are decoded and displayed on 8 LED indicators.

An RCA 1861 **video graphics** chip allows you to connect to your own TV with an inexpensive video modulator to do graphics and games. There is a **speaker system** included for writing your own music or using many music programs already written. The speaker amplifier may also be used to drive relays for control purposes.

A 24 key **HEX keyboard** includes 16 HEX keys plus **load, reset, run, wait, input, memory protect, monitor select** and **single step**. Large, on board displays provide output and optional **high and low address**. There is a 44 pin standard connector slot for PC cards and a 50 pin connector slot for the Quest Super Expansion Board. Power supply and sockets for **all** IC's are included in the price plus a detailed 127 pg. instruction manual which now includes over 40 pgs. of software info. including a series of lessons to help get you started and a music program and graphics target game. Many schools and universities are using the Super Elf as a course of study OEM's use it for training and R & D.

Remember, other computers only offer Super Elf features at additional cost or not at all. **Compare before you buy. Super Elf Kit $106.95, High address option $8.95, Low address option $9.95. Custom Cabinet** with drilled and labelled plexiglass front panel **$24.95.** Expansion Cabinet with room for 4 S-100 boards **$41.00. NiCad Battery Memory Saver Kit $6.95.** All kits and options also completely assembled and tested.

Questdata, a 12 page monthly software publication for 1802 computer users is available by subscription for $12.00 per year. Issues 1-12 bound $16.50.

Tiny Basic Cassette **$10.00,** on ROM **$38.00,** original Elf kit board **$14.95. 1802 software;** Moews Video Graphics **$3.50.** Games and Music **$3.00,** Chip 8 Interpreter **$5.50.**

Super Expansion Board with Cassette Interface $89.95

This is truly an astounding value! This board has been designed to allow **you** to decide how you want it optioned. **The Super Expansion Board comes with 4K of low power RAM** fully addressable anywhere in 64K with built-in memory protect and a **cassette interface**. Provisions have been made for all other options on the same board and it fits neatly into the hardwood cabinet alongside the **Super Elf**. The board includes slots for up to 6K of **EPROM** (2708, 2758, 2716 or TI 2716) and is **fully socketed.** EPROM can be used for the monitor and Tiny Basic or other purposes.

A 1K Super ROM Monitor $19.95 is available as an on board option in 2708 EPROM which has been preprogrammed with a program loader/editor and error checking multi file cassette read/write software, (relocatable cassette file) another exclusive from Quest. It includes register save and readout, block move capability and video graphics driver with blinking cursor. Break points can be used with the register save feature to isolate program bugs quickly, then follow with single step. The Super Monitor is written with subroutines allowing users to take advantage of monitor functions simply by calling them up. Improvements and revisions are easily done with the monitor. If you have the **Super Expansion Board** and **Super Monitor** the monitor is up and running at the push of a button.

Other on board options include **Parallel Input and Output Ports** with **full handshake**. They allow easy connection of an ASCII keyboard to the input port. **RS 232 and 20 ma Current Loop** for teletype or other device are on board and if you need more memory there are two **S-100** slots for static RAM or video boards. Also a 1K Super Monitor version 2 with video driver for full capability display with Tiny Basic and a video interface board. **Parallel I/O Ports $9.85, RS 232 $4.50, TTY 20 ma I/F $1.95, S-100 $4.50. A 50 pin connector set** with ribbon cable is available at $15.25 for easy connection between the **Super Elf** and the **Super Expansion Board.**

Power Supply Kit for the complete system (see Multi-volt Power Supply below).

Same day shipment. First line parts only. Factory tested. Guaranteed money back. Quality IC's and other components at factory prices.

INTEGRATED CIRCUITS

7400TTL	
7400N	.17
7402N	.18
7404N	.19
7409N	.23
7410N	.18
7414N	.70
7420N	.18
7422N	.39
7430N	.20
7442N	.50
7445N	.69
7447N	.60
7448N	.69
7450N	.18
7474N	.35
7475N	.49
7485N	.88
7489N	1.85
7490N	.43
7492N	.43
7493N	.43
7495N	.69
74100N	.90
74107N	.35
74121N	.34
74123N	.59
74125N	.45
74145N	.69
74150N	.95
74151N	.69
74154N	1.00
74157N	.69
74161N	.87
74162N	.87
74163N	.87
74174N	.96
74175N	.90
74190N	1.15
74192N	.87
74193N	.85
74221N	2.00
74298N	1.65
74365N	.66
74366N	.66
74367N	.66

74LS00 TTL	
74LS00N	.35
74LS02N	.35
74LS04N	.35
74LS05N	.35
74LS08N	.35
74LS10N	.35
74LS13N	.55
74LS14N	1.10
74LS20N	.35
74LS22N	.35
74LS28N	.41
74LS30N	.35
74LS33N	.75
74LS38N	.55
74LS74N	1.25
74LS75N	1.00
74LS90N	.85
74LS93N	.70
74LS95N	1.10
74LS107N	.45
74LS112N	.45
74LS113N	.35
74LS132N	.89
74LS136N	.45
74LS151N	.85
74LS155N	.85
74LS157N	.85
74LS162N	1.15
74LS163N	1.15
74LS174N	2.00
74LS190N	1.06
74LS221N	1.95
74LS258N	.67
74LS367N	1.35

LINEAR	
CA3045	.90
CA3046	1.10
CA3081	1.80
CA3082	1.90
CA3089	2.95
LM301AN AH	.35
LM305H	.87
LM307N	.35
LM308N	.89
LM309K	1.50
LM311H N	.90
LM317T K	3.75
LM318	1.35
LM320K 5	1.50
LM323K-5	5.95
LM320K-12	1.50
LM320K 15	1.50
LM320T-5	1.60
LM320T-8	1.60
LM320T 12	1.50
LM320T 15	1.60
LM324N	1.15
LM339N	1.55
LM340K-5	1.35
LM340K 8	1.35
LM340K 12	1.35
LM340K-15	1.35
LM340K 24	1.35
LM340T-5	1.25
LM340T-8	1.25
LM340T 12	1.25
LM340T-15	1.25
LM340T 18	1.25
LM340T-24	1.25
LM343H	4.50
LM350	7.50
LM370	1.15
LM377	3.00
LM379	5.00
LM380N	1.00
LM381	1.60
LM382	1.60
LM703H	.40
LM709H	.28
LM723H N	.50
LM733N	.67
LM741CH	.35
LM741N	.32
LM747H N	.75
LM748N	.35
LM1303N	1.75
LM1304	1.10
LM1305	1.27
LM1307	2.00
LM1310	2.75
LM1458	.47
LM1800	1.75
LM1812	7.50
LM1889	3.00
LM2111	1.75
LM2902	1.50
LM3900N	.60
LM3905	1.75
LM3909N	.89
MC1458V	.50
NE540H	3.75
NE550N	1.00
NE555V	.39
NE556A	.85
NE565A	1.00
NE566V	1.50
NE567V	1.00
NE570B	5.00
78L05	.60
78L08	.60
78M05	.85
75108	1.75
75491CN	.50
75492CN	.55
75494CN	.89

A to D CONVERTER	
8038B	4.50
8700CJ	13.95
8701CN	22.00
8750CJ	13.95
LD130	9.95
9400CJV F	7.40
ICL7103	9.50
ICL7107	14.25

CMOS	
CD34001 Fair	.50
CD4000	.16
CD4001	.28
CD4002	.28
CD4006	1.10
CD4007	.28
CD4008	.28
CD4009	.45
CD4010	.45
CD4011	.28
CD4012	.28
CD4013	.39
CD4014	1.00
CD4015	1.00
CD4016	.45
CD4017	1.05
CD4018	.94
CD4019	.45
CD4020	1.02
CD4021	1.25
CD4022	1.10
CD4023	.28
CD4024	.75
CD4025	.28
CD4026	2.00
CD4027	.66
CD4028	.85
CD4029	1.02
CD4030	.45
CD4035	1.02
CD4040	1.02
CD4042	.85
CD4043	.85
CD4044	.85
CD4046	1.67
CD4049	.45
CD4050	.49
CD4051	1.13
CD4060	1.42
CD4066	.71
CD4068	.40
CD4069	.40
CD4070	.50
CD4071	.28
CD4072	.28
CD4073	.28
CD4075	.28
CD4076	1.45
CD4078	.40
CD4081	.28
CD4082	.28
CD4116	.47
CD4490	5.50
CD4507	1.00
CD4508	4.25
CD4510	1.02
CD4511	.94
CD4515	2.52
CD4516	1.10
CD4518	1.02
CD4520	1.02
CD4527	1.51
CD4528	.79
CD4553	3.50
CD4566	2.25
CD4583	4.50
CD4585	1.10
CD40192	3.00
74C00	.28
74C04	.40
74C10	.28
74C14	2.10
74C20	.28
74C30	.28
74C48	1.95
74C74	.75
74C76	1.40
74C90	1.15
74C93	1.40
74C154	3.00
74C160	1.44
74C175	1.35
74C192	1.65
74C221	2.00
74C905	6.00
74C906	.75
74C914	1.95
74C922	5.50
74C923	5.50
74C925	6.95
74C926	6.95
74C927	6.95

INTERFACE	
8095	.65
8096	.65
8097	.65
8098	.65
8T09	1.25
8T10	4.50
8T13	3.00
8T20	5.50
8T23	3.10
8T24	3.50
8T25	3.20
8T26	1.69
8T28	2.75
8T97	1.69
8T98	1.69

MOS MEMORY RAM	
2101-1	3.95
2102 1	.95
2102AL-4	1.25
2102AN 2L	1.60
21L02 1	1.18
2114A-4	4.95
2107B-6	3.75
2111-1	3.75
2112 2	3.95
2114L 300ns	7.40
4116 200ns	9.50
8 4116 200ns	64.00
2513B	6.30
MM5262	.40
MM5280	3.00
MM5320	9.95
MM5330	5.94
PD411D-3	4.00
PD411D 4	5.00
P5101L	8.95
4200A	9.95
82S[illegible]5	2.90
91L[illegible]A	1.50
HD[illegible]165 5	6.95
MM57100	4.50
GIAY38500-1	9.95
MCM66751A	9.95
9368	3.50
410D	10.00
416	16.00

CLOCKS	
MM5311	5.50
MM5312	3.90
MM5314	3.90
MM5369	2.10
MM5841	14.45
MM5865	7.95
CT7001	5.80
CT7010	8.95
CT7015	8.95
MM5375AA N	3.90
MM5375AG N	4.90
7205	16.50
7207	7.50
7208	15.95
7209	4.95
DS0026CN	3.75
DS0056CN	3.75
MM53104	2.50

MICROPROCESSOR	
65[illegible]	12.50
6504	16.50
652[illegible]	13.60
6800	17.50
6802	18.75
6820	9.95
6850	12.95
8080A	8.95
8085	[illegible]7.00
8086	75.00
Z80	14.75
Z80A	19.75
8212	2.90
8214	8.00
8216	2.90
8224	2.90
8228	5.35
8251	8.50
8253	15.00
8255	9.25
8257	19.50
8259	19.50
1802CP plas	13.95
1802DP plas	17.95
1861P	11.50
CDP1802CD	19.95
CDP1802D	25.00
CDP1861	12.95

UART/FIFO	
AY5-1013	5.50
AY5-1014	7.50
3341	6.95

PROM	
17[illegible]A	3.95
25138 upper case	6.95
2708	7.75
2716T1	24.50
2716 Intel	34.95
8[illegible]716 Intel	240.00
2732	85.00
2758	2[illegible].50
8741A	85.00
8748	75.00
8748-8	70.00
8755A	65.00
N82S[illegible]3	2.95
N82S123	3.50
N82S126	3.75
N82S129	3.75
N82S131	3.75
N82S136	8.75
N82S137	8.75
DM8577	2.90
82[illegible]3	2.90
8[illegible]716 Intel	[illegible]40.00

CONNECTORS	
30 pin edge	2.50
44 pin edge	2.75
100 pin edge	4.50
100 pin edge WW	5.25

IC SOCKETS
Solder Tin Low Profile

PIN	1 UP	PIN	1UP
8	15	22	30
14	14	24	35
16	16	28	42
18	27	36	58
20	29	40	57

[illegible] level 14 pin ww .20

WIRE WRAP LEVEL 3

PIN		PIN	
14	32	24	86
16	33	28	1.00
18	57	40	1.23

CRYSTALS	
1 MHz	4.50
2 MHz	4.50
4 MHz	4.25
5 MHz	4.25
10 MHz	4.25
18 MHz	3.90
20 MHz	3.90
32 MHz	3.90
32768 Hz	4.0[illegible]
1.8432 MHz	4.50
3.5795 MHz	1.20
2.0100 MHz	1.95
2.097152 MHz	4.50
2.4576 MHz	4.50
3.2768 MHz	4.50
5.0688 MHz	4.50
5.185 MHz	4.50
5.7143 MHz	4.50
6.5536 MHz	4.50
14.31818 MHz	4.25
18.432 MHz	4.50
22.1184 MHz	4.50

KEYBOARD ENCODERS	
AY5-[illegible]376	$1[illegible].50
AY5-3600	17.95
AY5-91[illegible]	10.50
AY5-9200	16.50
74C922	5.50
74C923	5.50
HD0165-5	6.95

D Connectors RS232	
DB25P	2.95
DB25S	3.95
Cover	1.50
RS232 Complete Set	6.50
DE9S	1.95
DA15P	2.10
DA15S	3.10

Hickok 3½ Digit LED multimeter **69.95**
Stopwatch Kit **26.95**
Auto Clock Kit **17.95**

Not a Cheap Clock Kit $14.95
Includes everything except case 2-PC boards 6-.50" LED Displays 5314 clock chip transformer all components and full instructions Orange displays also avail **Same kit w/ .80" displays** Red only **$21.95** Case **$11.75**

8K 16K Eprom Kit (less PROMS) $89.00
Motherboard $39.00
Extender Board $8.99

RESISTORS ¼ watt 5%
10 per type .03 — 1000 per type .012
25 per type .025 — 350 piece pack
100 per type .015 — 5 per type 6.75
½ watt 5% per type .05

KEYBOARDS	
56 key ASCII keyboard kit	$67.50
Fully assembled	77.50
53 key ASCII keyboard kit	60.00
Fully assembled	70.00
Enclosure	14.95
Metal Enclosure	29.95

LEDS	
Red T018	.15
Green, Yellow T018	.20
Jumbo Red	.20
Green, Orange, Yellow Jumbo	.25
Clipite LED Mounting Clips	8/$1.25

(specify red, amber, green, yellow, clear)

CONTINENTAL SPECIALTIES in stock
Complete line of breadboard test equip.
MAX-100 8 digit Freq Ctr **$128.95**

OK WIRE WRAP TOOLS in stock
Portable Multimeter **$18.00**

SPECIAL PRODUCTS	
MM5865 Stopwatch Timer with 10 pg. spec	9.00
PC board	7.50
Switches Mom. Pushbutton	.27
3 pos. slide	.25
Encoder HD0165-5	6.95
Paratronics 100A Logic Analyzer Kit	**$224.00**
Model 10 Trigger Expander Kit	$229.00
Model 150 Bus Grabber Kit	$369.00
Sinclair 3½ Digit Multimeter	**$59.95**
Clock Calendar Kit	**$23.95**
2.5 MHz Frequency Counter Kit	**$37.50**
30 MHz Frequency Counter Kit	**$47.75**

TRANSFORMERS	
6V 300 ma	3.25
12 Volt 300 ma transformer	1.25
12.6V CT 600 ma	3.75
12V 250 ma wall plug	2.95
12V CT 250 ma wall plug	3.50
[illegible]4V CT 400 ma	3.95
10V 1.[illegible] amp wall plug	4.85
1[illegible]V 6 amp	12.95
12V 500 ma wall plug	4.75
12V 1 amp wall plug	6.50
12V 3 amp	8.50
10.15 VAC 8.16 VA wall plug	9.75

DISPLAY LEDS				
MAN1		CA	270	2.90
MAN3		CC	125	.39
MAN72 74	CA	CA	300	1.00
DL704		CC	300	1.25
DL707 DL707R		CA	300	1.00
DL727 728	CA	CC	500	1.90
DL747 750	CA	CC	600	1.95
DL750		CC	600	1.95
FND359		CC	357	.70
FND500 507	CC	CA	500	1.35
FND503 510	CC	CA	500	.90
FND800 807	CC	CA	800	2.20
3 digit Bubble				.60
4 digit Bubble				.80
DG8 Fluorescent				1.75
DG10 Fluorescent				1.75
5 digit 14 pin display				1.00
10 digit display				1.25
7520 Clairex photocells				.39
TIL311 Hex				9.50
MAN3640		CC	30	1.10
MAN4610		CA	40	1.20
MAN4640		CC	40	1.20
MAN4710		CA	40	.95
MAN4740		CC	40	1.20
MAN6640		CC	56	2.95
MAN6710		CA	60	1.35
MAN6740		CC	60	1.35
MA1002A				**8.95**
MA1002E				**8.95**
MA1012A				**8.95**
102P3 transformer				**2.25**

ROCKWELL AIM 65 Computer

6502 based single board with full ASCII keyboard and 20 column thermal printer. 20 char. alphanumeric display. ROM monitor, fully expandable. $375.00. 4K version $450.00. 4K Assembler $85.00, 8K Basic Interpreter $100.00.

Special small power supply for AIM65 assem. in frame **$49.00.** Complete AIM65 in thin briefcase with power supply **$485.00.** Molded plastic enclosure to fit AIM65 plus power supply **$47.50.** Special Package Price: 4K AIM, 8K Basic, power supply, cabinet **$599.00**

AIM65/KIM/VIM/Super Elf 44 pin expansion board; 3 female and 1 male bus. Board plus 3 connectors **$22.95.**

AIM65/KIM/VIM I/O Expansion Kit; 4 parallel and 2 serial ports plus 2 internal timers **$39.00.** PROM programmer for 2716 **$150.00.**

Multi-volt Computer Power Supply

8v 5 amp, ± 18v .5 amp, 5v 1.5 amp, −5v .5 amp, 12v .5 amp, −12 option. +5v, +12v are regulated. Kit $29.95. Kit with punched frame $37.45, $4.00 shipping. Kit of hardware $14.00. Woodgrain case $10.00, $1.50 shipping

PROM Eraser Will erase 25 PROMs in 15 minutes. Ultraviolet, assembled **$37.50**
Safety switch/Timer version **$69.50**

60 Hz Crystal Time Base Kit $4.40

Converts digital clocks from AC line frequency to crystal time base. Outstanding accuracy.

NiCad Battery Fixer/Charger Kit

Opens shorted cells that won't hold a charge and then charges them up, all in one kit w/full parts and instructions. **$7.25**

LRC 7000 + Printer $389.00

40/64 column dot matrix impact, std. paper Interface all personal computers

Televideo Terminal $845.00

102 key, upper, lowercase, 10 Baud rates 24 x 80 char. microprocessor cont. edit. cap.

Intertube II Terminal $874.00

Super Brain Floppy Disk Terminal $2895.00

79 IC Update Master Manual $29.95

Complete IC data selector, 2500 pg. master reference guide. Over 50,000 cross references. Free update service through 1979. Domestic postage $3.50. No foreign orders.

S-100 Computer Boards

8K Static RAM Kit	$135.00
16K Static RAM Kit	265.00
24K Static RAM Kit	423.00
32K Static RAM Kit	475.00
16K Dynamic RAM Kit	199.00
32K Dynamic RAM Kit	310.00
64K Dynamic RAM Kit	470.00
Video Interface Kit	**$129.00**

Video Modulator Kit $8.95

Convert TV set into a high quality monitor w/o affecting usage. Comp. kit w/full instruc.

Digital Temp. Meter Kit $34.00

Indoor and outdoor. Switches back and forth. Beautiful. 50" LED readouts. Nothing like it available. Needs no additional parts for complete, full operation. Will measure −100° to +200°F, tenths of a degree, air or liquid. Beautiful woodgrain case w/bezel **$11.75**

TERMS: $5.00 min. order U.S. Funds. Calif residents add 6% tax. BankAmericard and Master Charge accepted. Shipping charges will be added on charge cards.

FREE: Send for your copy of our NEW 1980 QUEST CATALOG. Include 28¢ stamp.

✓Reader Service—see page 257

CLASSIFIEDS

Classified advertisements are intended for use by persons desiring to buy, sell or trade used computer equipment. No commercial ads are accepted.

Two sizes of ads are available. The $5 box allows up to 5 lines of about 35 characters per line, including spaces and punctuation. The $10 box allows up to 10 lines. Minimize use of capital letters to save space. No special layouts allowed. Payment is required in advance with ad copy. We cannot bill or accept credit.

Advertising text and payment must reach us 60 days in advance of publication (i.e., copy for March issue, mailed in February, must be here by Jan. 1). The publisher reserves the right to refuse questionable or inapplicable advertisements. Mail copy with payment to: **Classifieds, Kilobaud Microcomputing, Peterborough NH 03458.** Do not include any other material with your ad as it may be delayed.

Wanted: SWTP computer with dual floppies. Dr. Gordon, 1435 W. 49th Place, Hialeah, Florida (305) 822-1100.

SOL 20 in original box complete with monitor and cassette recorder, $1495. 8K RAM board, $100. GPM PROM board, $60. Uses 2708 PROMs. Mountain Hardware 100.000 day clock, $100. Bob Haworth, Box 92, Boring OR 97009 or (503) 663-6206.

For Sale: Imsai 8080 mainframe with 22 slot motherboard, Imsai MIO board, 32K 250 ns memory, North Star minidisk and controller. Includes software for terminal and printer. Best offer. Call evenings at (305) 920-6545, or write to Overtone, Inc., 1443 Washington St., Hollywood FL 33020.

Ohio Scientific C2-4P: 24K extra RAM, baud rate sw, assembler, ext. monitor, 20 games, TV, extras. Works fine. Cost $1100. Sell for $800. F. Levi, Rte. 1, Box 67C, Hillsboro OR 97123. (503) 647-2609 (eve).

AIM-65 for sale w/4K RAM, power supply (5 V @ 8 A, 24 V @ 2 A), plus—current loop & audio cables. Mint cond. $400. Call (617) 997-2811, ext. 783, ask for Paul Smola.

TRS-80 Software based on Osborne disk payroll, accts payable, accts receivable and general ledger. Manuals included. Satisfaction guaranteed. Original cost, $500. Asking $325. Phone (312) 357-5827 after 6 PM.

For Sale: AIM-65 computer w/keyboard, *thermal printer*, 4K RAM, BASIC/editor/assembler in ROM. Retail, $600. Ask, $400 (or best). (303) 451-3193, 457-2488.

S-100 Xitan Z-80 w/32K RAM, North Star disk, 80 × 25 × 2 VDB Hitachi mon, Cherry kbrd, two RS-232s, one par. port. Cost $3300, Sell $1995. Paul, Bremerton WA. (206) 373-7626.

For Sale—Complete Heathkit Microprocessor Training Course w/ET-3400 trainer, ETA-3400 assy. w/3K chip set, and H9 terminal and all manuals. Completely assembled and tested. An $866 value for only $600. You save $266 from current Heath catalog. Contact Gene R. Barboza, GPO Box BT, San Juan PR 00936.

For Sale: Heathkit H9 terminal (with 24 lines), H8 computer, H8-5 serial I/O, *cassette recorder*, 16K of memory. All assembled and working. Tape operating software, all manuals and most REMarks to date included. $600. Steve Sama, 4006 Berrywood Dr., Seaford NY 11783. Work tel: (516) 293-4420.

For Sale: TI 722 30 cps ASCII thermal printer. Complete with manual, new printhead and ten extra circuit boards. Prints but needs some minor work; $150. Also have 32 memories, 4027 4K × 1 dynamic pullouts, includes schematic to expand TRS-80 memory; $1.50 ea. M. Bickerton, 3631 Wharton, Phila. PA 19146.

Sale: ASR-33 Teletype w/tape reader-punch, fully operational, could be interfaced to computer, w/documentation, w/stand, $450; w/o stand, $400; shipping extra. Call (803) 781-5647.

ELF II, ASCII keyboard, Giant Bd., 4K RAM bd., rf mod., exp. pwr supply, Tiny BASIC, light pen, all manuals & documentation. Asking $275. Bob Clark, 78 Riverside Ave., Merrick NY 11566.

Adds 580 terminal/keyboard 24 lines/80 characters, 12-inch screen uppercase, EIA and current loop 110-9600 band—used—working—manual. $450. Dan Henderson, (303) 545-5173.

For Sale: 2 National Multiplex 3M3-A digital cassette recorders and 2SIOR interface/monitor board. Storage capacity per 3M data cartridge of up to 2 megabytes. Search capability up to 50 kilobaud/sec. Will sell for best price over $275. B. Wagman, 2825 Lorcom Lane, Arlington VA 22207.

For Sale: TRS-80 Level II with 48K RAM, expansion interface, 2 disk drives, lowercase mod and Electric Pencil. New DOS, CP/M, C-BASIC, Selecta III, light pen, etc. Asking $2200. Call days (212) 477-2841.

SS-50 bus (6800) F&D Assoc. VDB-1 video board, 64 char × 16 lines, scrolling, screen erase, reverse video, etc. Expertly assembled and socketed, not tested. With complete documentation; $75. Also (2) 4K SWTP memory boards—socketed, expertly assembled & working; $75 ea. Paul Ramos, (617) 935-3758 after 6 PM.

For Sale—Digital group computer system. 50K mem., 4 Phi decks, monitor, keyboard, software. Up and running. Must sell. Paul Bachman, 11705 Munson Hwy., Morenci MI 49256. (517) 239-2972.

Will buy 8" floppy drives. State make, condition, price. Bob Guest, PO Box 880, Duncan OK 73533.

For Sale: 4 Proc. Tech. 16K dynamic RAM boards, $125 ea. CP/M-ISIS conv. utility, $60. COM-to-HEX conv., $30. Rick Odle, 16919 Camberwell Green Lane, Houston TX 77070.

For Sale: SWTPC 6800/2 computer syst. Includes SWTPC 6800/2 computer with SWTBUG monitor, AC-30 cassette interface, 20K RAM memory, and CT-64 terminal without monitor. Free software includes 8K BASIC, 4K BASIC, assembler disassembler, program relocator, Star Trek and more. $650. Write to Michael W. Taylor, PO Box 3823CRS, Johnson City TN 37601.

Elf II, Giant, 4K (2W instld), L pen, case, Elfbug, T-BASIC, RCA man, S course, software—$200. Radio Shack ASCII kybd w/cust encl—$50. Taylor, 1200 Mkt. St., Bloomsburg PA 17815. (717) 784-6173.

For Sale: Vector Graphic I + + Mainframe with 18 A 8V, 2.5 A + −16 V P.S., 18-slot motherboard, cutouts for two minifloppy drives. Scott Bishop, 2221 Charlotte Drive, Maitland FL 32751. Phone (305) 869-4203.

TRS-80—32K—expansion interface—quick printer II. Best offer. S. Dietz, 6 Bush St., Newport RI 02840.

For Sale: Heath WH-14 line printer, assembler, used less than 10 hours. Also have Small Systems adapter to permit use on TRS-80, TTY, etc. Both for $725 or best offer. Call collect, (716) 838-4957 or write J. Anthony Greaves, 7144 Northview Dr., Lockport NY 14094.

Two SWTP 6800 mainframes, 28K, 1024 terminal, CT-64 terminal and monitor. Percom LFD 400, 90kb floppy, PR-40 printer, two AC-30s, 8K Level 2.3 BASIC. As unit or parts. Offers? Dave, Creative Computing, Mackenzie Mall, Victoria, BC, Canada. (604) 477-7732.

For Sale: Compucolor II, 8K RAM, 16K ROM. About 15 games on 10 disks. New, in good working order. 8-color monitor. $1100 certified check, I'll pay shipping. Also, much TRS-80 software. Database for 2 disk, 48K system. Allows use of printer. Handles 100 items, each w/15 fields. Sorts, updates by percentage, easy editing. 4 different programs. Allows form printout in easy format. $75. Sandy Sigal, 6851 Mammoth Ave., Van Nuys CA 91405. (213) 989-5488.

Tutorial Computer Programs Wanted for a book of instructional programs in BASIC which science teachers may use with microcomputers for enrichment or remediation. For information on submitting programs or to be notified when the book is ready, please write to Theodore Wade, 106 Hodges Ln., Takoma Park MD 20012.

Apple II computer for sale, new, disk memories, printer, modem, B&W mon., Pascal card, firmware board, programmers aid ROM, autostart ROM. Sell all or part. (714) 776-6384, Don.

Computer System—Imsai with 64K memory—2 MITS floppy disks—Hazeltine 1500 CRT and Heath Lineprinter, excellent condition, just like new. $6000. Memphis TN (901) 754-6784.

For Sale: SD Systems Z-80 Starter Kit a&t with manuals, $300 or best offer. Gregory Overby, Qtrs. 6458F, USAFA CO 80840. (303) 472-1764.

North Star single-density disk controller and software, $250. Includes S-100 controller board, power regulation board for drive, all cables, DOS, N.S. Extended BASIC and all manuals. Requires at least 1 Shugart SA400 drive (will handle 3) and 16K mem. at 2000H. Has Horizon I/O or will customize to your machine for $5 if you supply info. Call Ken (919) 766-8046 in PM.

FOR TRS-80: Comprint 912 printer 80 column, 225 cps, parallel, w/6' cbl., $540. Vista V80 40 trk w/NEWDOS + & cable, $385. Exp. Int. w/32K, $425. Compumax Software A/P, A/R, Inv, GL, PR., $300. Sell all for $1450. Unmarked 2114L2 or 2141L2 $4.50 ea. Unmarked 2114L4 or 2141L4 $3.50 ea. Richard Smith, 3648 Madrid Dr., SJ CA 95132. (408) 946-0735.

Computer Portrait System: SWTPC 6800 w/16K, Smartbug, CT-1024, Microworks Digisector & PROM Board, Malibu 160 matrix printer, Toshiba zoom video camera, 2 monitors, complete software in PROM. Will take best offer for totally integrated system or components. Also: SSB BFD-68 disk system ($675) or best offer, SWTPC AC-80 ($65), TASA touch keyboard ($80 new), plus asst. software. A. Eng, Box 238, Henrietta NY 14467. (716) 334-5162, (716) 422-4150.

For Sale: Heath H9 video term, $395. Heath SB-100 amateur trans. Both in mint operating condition. Write. Gerry VanLoh, Rte. 1, Box 416A, Sioux Falls SD 57101.

SWTP 4K memory boards. Low power chips —fully socketed. Only 5 left! Send $50 to: Ted Wolff, 579 W. 215th St., NY NY 10034.

"What are you complaining about? You're the one who asked it to rate your sex appeal!"

Power Supplies! Power Supplies! Power Supplies! SOLID STATE!! (5)

We got 'em! Take your pick . . .

These units are ideal for micro computers. They have been removed from equipment, checked out and guaranteed.

1—5 volts @ 8 amps + 12 volts @ 2 amps + 6 volts @ 75 MA. Power supply has a 3-wire line cord and fused. Dimensions: 10½" × 5½" × 4½". Shipping weight: 16 lbs. . . . . 37.50 ea. 2/70.00

2—Model 818, 5 volts at 15 amps + 12 volts at 4 amps-12 volts at 2 amps. (with line cord). . . . . 35.00 ea. 2/65.00

3— + 5 volts at 5 amps ± 12 volts at 500 ma. + 6 volts at 25 ms. (line cord included). . . . . 32.95 ea. 2/60.00

4—Elexon, multi output. Input: 120/240 AC, ±10%, 47-63 hz; output: 1) 12V, 1.5A, DC, OVP; 2) 12V, 1.5A, D.C., OVP. New, in box with operating instructions. 31.50

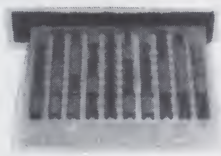
5—Power Design, Model 1210, constant voltage, DC. P.S. input: 105-125 A.C., 55 to 440 hz. Output: 1-12 volts, 0-10 amps, DC. continuously adjustable output voltage and current limiting. 139.00

COMPUTER GRADE CAPACITORS . . .

18,000 mfd 10 VDC	1.25	11,000 mfd 25 VDC	1.50	4,000 mfd 75 VDC	1.75
4,400 mfd 20 VDC	1.00	35,000 mfd 35 VDC	3.50	1,000 mfd 100 VDC	1.00
46,000 mfd 20 VDC	2.50	10,000 mfd 50 VDC	2.50	6,800 mfd 100 VDC	3.50
3.000 mfd 25 VDC	1.00	22,000 mfd 60 VDC	3.75	4,700 mfd 150 VDC	3.75

WIRE WRAP BOARDS

These boards are pre-wired and removed from equipment. Easy to unwrap for setting up your own board, contains mostly 14-pin IC sockets with individual pin connections. Each board has VCC and ground planes.



Smaller board measures 6½" × 6" and has 40 to 50 sockets.
Larger board measures 13½" × 6" and has 75 to 100 sockets.

Reduced prices **$7.50 ea. 2/$14.00**
$12.50 ea. 2/$23.00

DIABLO System Disc Drive

SERIES 40, MODEL 43

100 tracks per inch, total capacity of 50 megabits, w/Model 429 power supply, sector counter, 24 sectors, 1 fixed disc, 1 removable disc, average access time 38 ms, PPM: 2600, dimensions: 10 5/16" high, fits in standard rack, equipped with full extension slides, excellent used condition. Shipped freight collect.

$2495

HEWLETT PACKARD model 200CD/rack mounted AUDIO OSCILLATOR freq:5hz to 600khz output: 160mw $165.00

HEWLETT PACKARD model 400D ANALOG VACUUM TUBE VOLTMETER freq: 10hz to 4mhz voltmeter range: 1mv to 300vac in 12 ranges $85.00

TRANSFORMERS

ISOLATION STEP-DOWN TYPE

Primary: 230/115V, 50/60 CPS, Secondary: 115 volts output 250 VA.

$13.95 EACH

ROTRON WHISPER FANS

Unused, Model Rotron MU 3A1, 230V, AC, 14 watts, 50/60 hz, guaranteed, 4½" × 4½" × 1½"

$8.95

Clock Crystal Oscillators—TTL, Vectron, type CO-231T. Crystal freq. 4.9152 mhz. Input voltage 5 VDC ±. Output: Drives 10 TTL Loads Logic "0": 0.4V max., sink 16ma. Logic "1" 2.4V *min source 2 ma.* (above 50 mhz drives 2 Schottky TTL loads). Tuning adjust. with nominal range of ±30 ppm below 25 mhz and 15 ppm above 25 mhz. R.F.E. 1½" × 1½" × ½". $13.95

SG-132 SWEEP SIGNAL GENERATOR

FREQ: 15 TO 400 MHZ, VHF-UHF

Output: AM & FM: CW. FM deviation: ±1% to ±20% at any frequency. Crystal markers every 200Khz, 1mhz, 5mhz or ±10B. Frequency accuracy ±1%. Built-in oscilloscope for observing waveforms.

$329

TRENDLINE PHONES

Manufactured by I.T.T.

These units have rotary dials. Colors are: white, black, red, and green. They are packaged and have 6-foot cord and installation instructions. Used, but in good operating condition.

34.50 WALL TYPE

Minimum order $25.00. Items offered subject to prior sale. FOB, Brockton, Mass. Money order or check w/order. Shipments and handling add 5%. Shipments by parcel post or UPS. No CODs. Mass. residents add 5% sales tax.

WALLEN

ELECTRONICS CO. INC. Tel: (617) 588-6440-6441
✓45 108 SAWTELL AVE., BROCKTON, MA. 02402

ELECTRONIC COMPONENTS TEST EQUIPMENT CONNECTORS—WIRE

✓Reader Service—see page 257

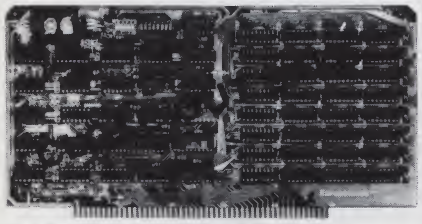
JADE Computer Products

SEND FOR OUR NEW 1980 CATALOG TODAY

OUR BEST SELLING MEMORY

EXPANDORAM

EXPANDABLE TO 64K USING 4116 RAMS



Interfaces with most popular S-100 boards
Bank selectable; PHANTOM provision
Draws only 5 watts fully populated
Designed to work with Z-80, 8080, and 8085 systems
No wait states required
16K boundaries & protect via dip switches
Kits come with sockets for full 64K
Invisible refresh

MEM-16130K (16K KIT) **$239.95**
MEM-16130A (16K A&T) **$289.95**
MEM-32131K (32K KIT) **$309.95**
MEM-32131A (32K A&T) **$359.95**
MEM-48132K (48K KIT) **$379.95**
MEM-48132A (48K A&T) **$429.95**
MEM-64133K (64K KIT) **$449.95**
MEM-64133A (64K A&T) **$499.95**

SOLID STATE MUSIC

PB-1

EPROM PROGRAMMER FOR 2708 OR 2716

MEM-99510K(KIT) $125.00
MEM-99510A(A & T) $175.00

VECTOR GRAPHICS

HI-RES GRAPHICS

256X240 S-100 HI—RES GRAPHICS BOARD

IOV-1070K(KIT) $150.00
IOV-1070A(A & T) $199.95
LIMITED TO STOCK ON HAND

SOLID STATE MUSIC

I/O-4

2 SERIAL & 2 PARALLEL I/O PORTS

IOI-1010K(JADE KIT) $149.95
IOI-1010A(A & T) $199.95

S D SYSTEMS

SBC-100/200

2 OR 4 MHz SINGLE BOARD COMPUTER

S-100 bus compatible Z-80 CPU
1K of on-board RAM
4 EPROM sockets accomodates 2708, 2716, or 2732
One parallel and one serial I/O port
4-channel counter timer chip (Z-80 CTC)
Software programmable serial baud rates

CPC-30100K (2 MHz KIT) **$249.95**
CPC-30100A (2 MHz A&T) **$299.95**
CPC-30200K (4 MHz KIT) **$289.95**
CPC-30200A (4 MHz A&T) **$339.95**

GET THE INSIDE TRACK

JADE DOUBLE-D

DOUBLE DENSITY DISK CONTROLLER

Read/write single or **double density,** 8" or 5¼" drives
On board Z-80 insures reliable operation
CP/M compatible in either single or double density
Density is software selectable
Up to 4 single or double sided, single or double density drives may be mixed on the same system
EIA level serial printer interface on board-up to 9600 baud (perfect for despooling operations)
All the hard work of disk access is done by the on board Z-80A and 2K memory, leaving your host CPU free for its normal duties
Uses IBM standard formats for proven reliability
THIS BOARD REALLY WORKS !!!!!!

IOD-1200K (DOUBLE-D KIT) **$285.00**
IOD-1200A (DOUBLE-D A&T) **$349.00**
IOD-1200D (MANUAL ONLY) **$15.00**

S D SYSTEMS

VERSAFLOPPY II

DOUBLE DENSITY DISK CONTROLLER

Single or double density floppy disk controller
985600 bytes on 8" double sided diskettes
259840 bytes on double sided 5¼" diskettes
S-100 bus (IEEE) standard compatible
IBM 3740 format in single density
8" and 5¼" drives controlled simultaneously
Operates with Z-80, 8080, and 8085 CPU's
Controls up to 4 drives
Vectored interrupt operation optional

IOD-1160K (KIT) **$335.95**
IOD-1169A (A&T) **$385.95**

S D SYSTEMS

VERSAFLOPPY

VERSATILE FLOPPY DISK CONTROLLER

IBM 3740 soft sectored format
S-100 Z-80 or 8080 compatible
Controls up to 4 single or double sided drives
Compatible with all popular disk drives
CP/M compatible
Listings for control software included

IOD-1150K (KIT) **$189.95**
IOD-1150A (A&T) **$239.95**

NEW DISK OPERATING SYSTEM

SDOS IS HERE

WE THINK IT'S SUPERIOR

SDOS is a CP/M compatible operating system designed for the S.D. Sales Versafloppy I or II. It requires the SBC-100/Versafloppy board set and functions as a superset of CP/M, giving 19 additional functions including file attributes, disk label, and read/write logical blocks. It provides additional protection features, and is expandable to a multi-user real-time system.
SDOS sells for $200.00

CALL FOR SALE PRICES !!!

32K STATIC RAM

EXPANDABLE 8K/32K, 2/4MHz, KIT/A&T

NEW 2 OR 4 MHz REV. C BOARD

THE JADE BIG Z

Z-80 CPU BOARD WITH SERIAL I/O PORT

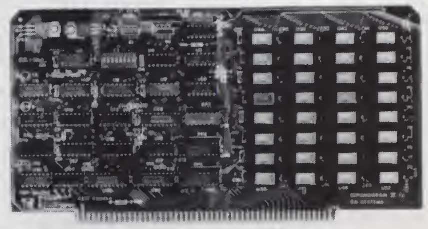
2 or 4 MHz switchable, on-board 2708, 2716, or 2732 EPROM useable in SHADOW mode (full 64K RAM)
Automatic MWRITE generation if no front panel
On-board USART for sync or async RS232

CPU-30201K (KIT) **$159.00**
CPU-30201A (A & T) **$209.00**

S D SYSTEMS

EXPANDORAM II

4 MHz RAM BOARD EXPANDABLE TO 256K



S-100 bus compatible, up to **4 MHz** operation
Expandable memory from 16K to 256K
Dip switch selectable boundaries
Page-mode allows up to 8 boards on the same bus
Invisible refresh; PHANTOM output disable
Designed to operate in Z-80 based systems

MEM-16631K (16K KIT) **$295.95**
MEM-16631A (16K A&T) **$345.95**
MEM-32632K (32K KIT) **$369.95**
MEM-32632A (32K A&T) **$419.95**
MEM-48632K (48K KIT) **$444.95**
MEM-48632A (48K A&T) **$494.95**
MEM-64632K (64K KIT) **$519.95**
MEM-64632A (64K A&T) **$569.95**

VECTOR GRAPHICS

12K PROM/RAM

1K OF STATIC RAM AND 12 2708 SOCKETS

MEM-12340A(A & T, NO PROMS) **$135.00**

BEST BUY RATED

TV-1 ONLY $7.95

OUR BEST SELLING R. F. MODULATOR

GET YOUR "ROCKS" OFF..THE SHELF

CRYSTAL SALE

WE STOCK 40 STANDARD FREQ. 3 FOR $12

TIMELY SPECIAL

60 Hz TIME BASE

CRYSTAL CONTROLLED KIT ONLY $4.95

COMING SOON

NEW JADE P/S I/O

PARALLEL/SERIAL/INTERRUPT BOARD

Z-80 SIO/PIO, 2 CTCs, expands to 2 SIOs, 4 CTCs
4 serial ports (async, sync, bisync, SDLC/HDLC)
2 parallel ports **with full handshake**
Software baud rate generators, interval timers, counters, and generates 32 vectored interrupts
Designed especially for MP/M multi-user multi-tasking operating systems. For use with Z-80 only

IOI-1045B (BARE BOARD) **$45.00**
IOI-1045K (KIT) **$169.95**
IOI-1045A (A & T) **$224.95**

SALE SALE SALE SALE SALE

EPROMS

BUY NOW AND SAVE

1702A $4.95
2708 .... $8.95
2516 .... $34.95
2716(T.I.) $34.95
2716(5 VOLT) $34.95
2758 .... $34.95
2532 .... $74.95
2732 .... $74.95
CALL FOR PRICING ON LARGE QUANTITIES

PRICES *SLASHED* FOR APRIL !

CALL TOLL-FREE AND SAVE

800-421-5809 CONTINENTAL U.S.   800-262-1710 INSIDE CALIFORNIA

TERMINAL SALE

ADDS REGENT

OUR FINEST LINE OF TERMINALS

ADDS REGENT 20	$795.00
ADDS REGENT 25	$850.00
ADDS REGENT 40	$1195.00
ADDS REGENT 60	$1495.00

SPECIAL PACKAGE PRICE

RS-232 SET $6.50

1 MALE DB-25, 1 FEMALE DB-25, 1 COVER

MICRO WORKS

2708 PROGRAMER

FITS STANDARD SWPT 6800 I/O SLOT

LIMITED SUPPLY(A & T) $75.00

VECTOR GRAPHICS

S-100 CARD CAGE

17" RACK WITH 18 SLOT MOTHERBOARD

ENC-118100(A & T) $210.00

SPECIAL CLEARANCE SALE

4K EPROM BOARD

HOLDS UP TO 16 INEXPENSIVE 1702A's

MEM-04220K(KIT NO PROMS)	$45.00
MEM-04220A(A & T NO PROMS)	$79.95

CP/M 2.0

Digital Research has done it again! This new release of their industry standard disk operating system is bound to be an even bigger hit than the original version. All of the fundamental file-size restrictions of release 1 have been eliminated, while maintaining full compatibility with the earlier versions. This new release can be field-configured by the user for a single mini-disk up through a multiple drive hard-disk system with 128 megabyte capacity. Field configuration can be accomplished easily through use of the Macro Library (DISKDEF) provided with CP/M 2.0.

A powerful operating system for only ... $150.00

JADE'S NEW MOTHERBOARDS

THE ISO-BUS

WE'RE PROUD OF OUR MOTHER !

6-SLOT	
BARE BOARD	$24.95
KIT	$49.95
ASSEMBLED & TESTED	$59.95
12-SLOT	
BARE BOARD	$39.95
KIT	$89.95
ASSEMBLED & TESTED	$99.95
18-SLOT	
BARE BOARD	$59.95
KIT	$129.95
ASSEMBLED & TESTED	$149.95

SPECIAL PACKAGE PRICE

ROCKWELL AIM-65

THE HEAD-START IN MICROCOMPUTERS

KIM-1 compatible
On-board printer
Full ACSII keyboard

AIM-65 w/1K RAM	$375.00
AIM-65 w/4K RAM	$450.00
8K BASIC ROM	$100.00
POWER SUPPLY	$59.95
CASE for AIM-65	$49.95
4K Assembler/Editor	$80.00

Special Package Price $599.00

4K AIM-65, 8K BASIC ROM, Power Supply, and Case

JADE

MEMORY EXPANSION KITS

FOR

TRS-80 APPLE EXIDY

Everything you need to add 16K of memory to your computer. Your kit comes neatly packaged with easy to follow instructions. In just minutes your computer is ready to tackle more advanced software.

$69.00

SALE SALE SALE SALE SALE

TRANSISTORS

LOOK UP THE SPECS AND STOCK UP

LIMITED TO STOCK ON HAND

2N456A	10/$1.00
2N718	10/$1.00
2N2222A	5/$1.00
2N3392	10/$1.00
2N3741	2/$3.00
2N3904	8/$1.00
2N4013	20/$1.00
2N4348	3/$5.00
2N5088	5/$2.00
2N5434	5/$1.00
2N5449	5/$1.00
2N6569	4/$1.00
3N81	5/$1.00

MICROPROCESSORS	
F8	$16.95
Z80 (2MHz)	$10.95
Z80A (4MHz)	$14.95
CDP1802CD	$24.95
6502	$11.95
6800	$12.50
6802	$20.00
8008-1	$15.95
8035	$24.00
8035-8	$24.00
8080-A	$10.00
8085	$23.00
TMS9900TL	$49.95
8080A SUPPORT DEVICES	
8212	$5.00
8214	$4.65
8216	$2.95
8224 (2MHz)	$4.30
8226	$2.75
8228	$6.40
8238	$6.40
8243	$8.00
8251	$7.50
8253	$20.00
8255	$6.40
8257	$19.95
8259	$19.95
8275	$69.95
8279	$17.70
USRT	
S2350	$10.95
UARTS	
AY5-1013A	$5.25
AY5-1014A	$8.25
TR1602B	$5.25
TMS6011	$5.95
IM6403	$9.00
BAUD RATE GENERATORS	
MC14411	$10.00

6800 PRODUCT	
6821P	$5.25
6828P	$12.00
6834P	$16.95
6850P	$4.80
6852P	$7.50
6860P	$9.25
6862P	$12.00
6875L	$7.30
6880P	$2.50
CHARACTER GENERATORS	
2513 Upper	$7.95
2513 Lower	$6.75
2513 Upper (5 volt)	9.75
2513 Lower (5 volt)	$13.00
MCM6571 up scan	$13.00
MCM6571A down scan	$10.95
PROMS	
1702A	$5.00
2708	$8.95
2716	$39.95
2716 (5v)	$39.95
2758 (5v)	$30.00
DYNAMIC RAMS	
4116/416D	8 for $74.95
2104/4096	$4.75
2107B-4	$3.95
TMS4027/4096	$4.75
STATIC RAMS	
21L02 (450ns)	$1.50
21L02 (250ns)	$1.75
2101-1	$2.95
2111-1	$3.25
2112-1	$2.95
2114L (450ns)	$5.75
2114L (300ns)	$5.95
TMS4044 (450ns)	$8.00
TMS4044 (300ns)	$9.95
410D (200ns)	$9.95
4200A (200ns)	$9.95

JADE ✓48

Computer Products

4901 W ROSECRANS, HAWTHORNE, CA 90250

213-679-3313

PLACE ORDERS TOLL FREE

800-262-1710 INSIDE CALIFORNIA

800-421-5809 CONTINENTAL U.S.

WRITE FOR OUR FREE 1980 CATALOG

FOR CUSTOMER SERVICE OR TECHNICAL INQUIRIES CALL 213-679-3317

TERMS OF SALE: Cash, checks, money orders, and credit cards accepted. **Minimum order $10.00.** California residents add 6% sales tax. Minimum shipping and handling charge $2.50. Prices are for U.S. and Canadian delivery only and are subject to change without notice. For export prices and information send for a *JADE INTERNATIONAL CATALOG.*

master charge · AMERICAN EXPRESS · VISA

INTEGRAL DATA SYSTEMS

THE PAPER TIGER

132 COLUMN DOT MATRIX PRINTER

Up to 198 CPS
1.75 to 9.5 inch adjustable tractor and friction feed.
Parallel and serial interface.
98 character ASCII set
80 to 132 columns.
6 or 8 lines per inch.
Eight software selectable character sizes.
110, 300, 600, or 1200 baud.

PRM-33440	$950.00
PRM-33441(GRAPHICS & 2K BUFFER)	$1050.00

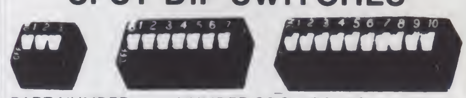
GUARANTEED PREMIUM QUALITY

JADE DISKETTES

MAGNIFICENT MAGNETIC MEDIA !

5¼" single sided, single density, box of 10	
MMD-5110103 (SOFT SECTOR)	**$29.95**
MMD-5111003 (10 SECTOR)	**$29.95**
MMD-5111603 (16 SECTOR)	**$29.95**
5¼" double sided, double density, box of 10	
MMD-5220103 (SOFT SECTOR)	**$39.95**
8" single sided, single density, box of 10	
MMD-8110103 (SOFT SECTOR)	**$34.95**
8" single sided, double density, box of 10	
MMD-8120103 (SOFT SECTOR)	**$55.95**
8" double sided, double density, box of 10	
MMD-8220103 (SOFT SECTOR)	**$57.95**

SPST DIP SWITCHES

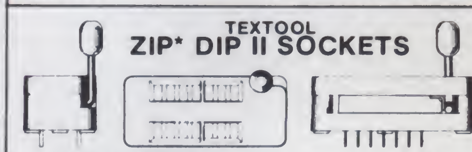


SALE

PART NUMBER	NUMBER OF SWITCHES	PRICE	
SWD-103	3	$1.00	~~$1.18~~
SWD-104	4	$1.05	~~$1.20~~
SWD-105	5	$1.10	~~$1.24~~
SWD-106	6	$1.15	~~$1.26~~
SWD-107	7	$1.20	~~$1.30~~
SWD-108	8	$1.25	~~$1.34~~
SWD-109	9	$1.30	~~$1.36~~
SWD-110	10	$1.35	~~$1.38~~

TEXTOOL

ZIP* DIP II SOCKETS



16 PIN ZIP* DIP II	$5.50
24 PIN ZIP* DIP II	$7.50
40 PIN ZIP* DIP II	*10.25*

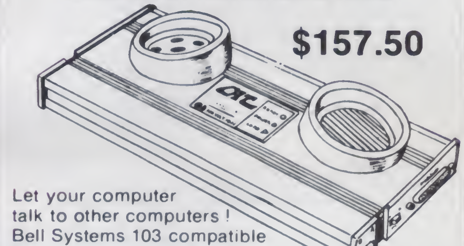
* ZERO INSERTION PRESSURE

SPECIAL PRICE FOR APRIL

NOVATION CAT

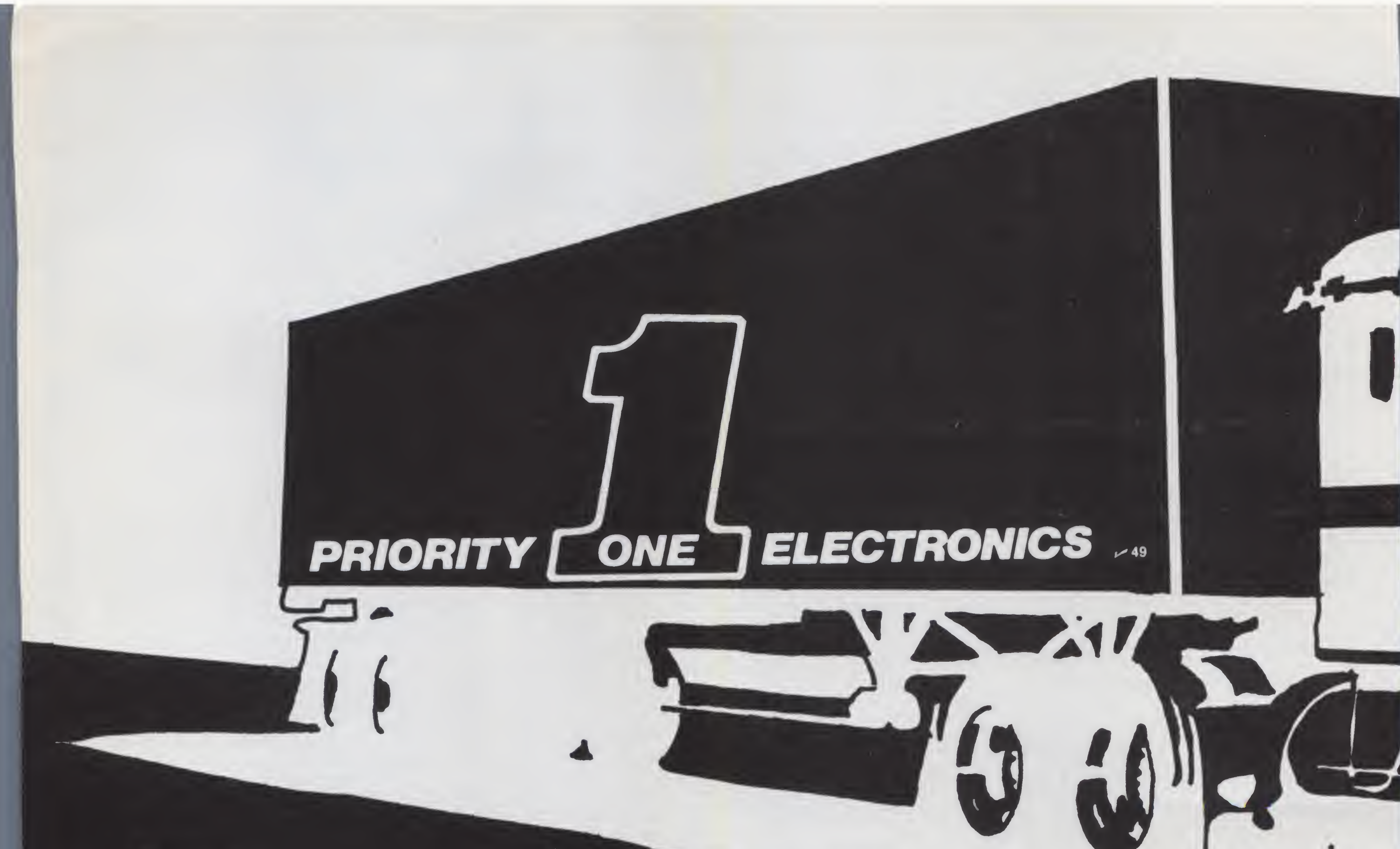
ACOUSTIC COUPLER/MODEM

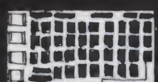
$157.50



Let your computer talk to other computers !
Bell Systems 103 compatible
300 baud, answer or orginate

IOM-5200A (SALE PRICED) $157.50





CCS-M-XVI-M

The true 16K Static Ram module for S-100 bus systems. ASSEMBLED & TESTED--100% BURN IN - FEATURES: • Fully static • Uses popular 2114 static RAMS • +5 volt operation only • Bank Select available by bank port and bank byte • Phantom Line capability • Addressable in 4K blocks • 4K blocks can be addressed anywhere with in 64K in 4k increments. • Meets IEEE proposed S-100 signal standards • LED indicators for board selection and bank selection • FR-4 EPOXY PC boards • Solder masked on both sides • Silk screen of part number and part designator.

OUR PRICE

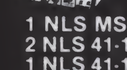
CCS-2016BA 450ns 2MHZ -
Reg. Price $349.95 . . . . $279.95
CCS-2016BB 300ns 4MHZ -
Reg. Price - $389.95 . . . . $309.00
CCS-2016BY Bare Board - only . . $29.95

The VISTA V-80 Disk Drive System

• 23% more storage capacity than TRS-80 • 120 day warranty • 40 track *patch at NO* CHARGE from VISTA

Single drive system $395.00
Two drive system $770.00
Four drive system $1450.00
Two drive cable $29.95
Four drive cable $39.95

NEW MS-230 DUAL TRACE



MINISCOPE 30 MHz BANDWIDTH

1 NLS MS-230 30 MHZ Scope . . . $598.15
2 NLS 41-141 10 to 1 Combo Probe$54.00
1 NLS 41-180 Deluxe Leather Case$45.00

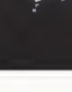
LIST PRICE $697.15
MS 230 COMBO PRICE $547.15
SAVE $150.00

AUDIO RESPONSE™ DIGITAL DISPLAY

The ROAD RUNNER offers Audio Response...A new dimension in multimeters. Rapid checking of fuses, capacitors, diodes, resistors, transistors, LED's darlingtons...without looking at the meter. The <1V and<2V ranges cause sufficient voltage across the probe tips to forward bias diodes and transistors causing a continuous tone. Similarly, capacitors can be charged until the tone shuts off...thus the approximate value of the capacitor can be measured. The <1 ohm,<10 ohm, and<100 ohm ranges are used in threshold testing of resistors and making continuity tests of PCB's, wire wrap and solder connections, light bulbs, and transformer windings. In addition, the Audio Response function permits the monitoring of incoming digital pulses. The ROAD RUNNER ADMM ™ FEATURES • Six Functions • 29 Ranges • 0.5% Accuracy on DCV • 5 Range Audio Response Function • Color coded easy-to-read front panel and pushbuttons • Full Accessory Line Expands Capability • 0.5" LCD Display • Rugged Case for "Field Use" • RFI Shielded.

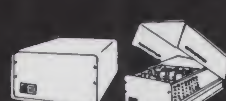
ONLY $139.00

WESTON-6100
Digital Multimeter . . . . @ $139.00
WESTON-6100C Leather Case @ $ 15.95
WES-9740-0279627 VHF RF
Voltage Probe . . . . @ $ 63.00
WES-5768 AC Clamp-on Probe @ $ 57.75
WES-1961 RF Voltage Probe . . @ $ 29.00
WES-9395 50 KV Probe . . . . . . @ $ 38.75
WES-6100-AC 110V
AC Adapter . . . . @ $ 15.00



NEW FROM SAMS BOOKS

192 PAGES TRS-80 . . . . $8.95



CALIFORNIA Computer Systems

• S-100 compatible • Industrial/commercial quality construction • Flip-top cover • Excellent cooling capability • 12 slot capability (uses model 2501A) • Input 105, 115, or 125 VAC • Output + 8 VDC20A, + – 16 VDC 4A • Active termination of all bus lines • Fan and circuit breaker included • Rugged construction

CCS-2200A Assembled & Tested
35 lbs $399.95
CCS-2200AK Kit 35 lbs. $349.95

MEMORY MEMORY

2102LIPC
Low Power 450ns in lots of 20 . . . . $1.10
2102AL-2
Low Power 250ns in lots of 20 . . . . $1.25
2114-3L
1Kx4 300ns Low Power . . . . 8/$50.00
5257-3L
4Kx1 300ns Low Power . . . . . . 8/$55.00
2708
8K 450ns EPROM . . . 8/$60.00-$8.50 ea.
2716
16K 5 Volt only EPROM
8/$248.00 . . . . $32.00 ea.

RS232 and "D" SUB-MINIATURE CONNECTORS *

P Plug, Male Type S Socket, Female Type C Cover Hood

PART NO.	DESCRIPTION	PRICE 1-9	10-24	25-99
CND-DE9P	9 Pin Male	1.60	1.40	1.30
CND-DE9S	9 Pin Female	2.25	2.00	1.90
CND-DE9C	9 Pin Cover	1.50	1.35	1.20
CND-DA15P	15 Pin Male	2.35	2.15	2.00
CND-DA15S	15 Pin Female	3.25	3.10	2.90
CND-DA15C	15 Pin Cover	1.60	1.45	1.30
CND-DB25P	25 Pin Male	2.80	2.60	2.40
CND-DB25S	25 Pin Female	3.60	3.40	3.20
CND-DB51212-1	1 pc.Grey Hood	1.50	1.30	1.10
CND-DB110963-3	2 pc.Grey Hood	1.75	1.50	1.35
CND-DB1226-1A	2 pc.Black Hood	1.90	1.65	1.45

. . . Terminals and Printers

TELEVIDEO TVI-912

Upper case and lower case; 15 baud rates: 75 to 19,000 baud; dual intensity; 24 x 80-char. display, 12 x 10 resolution. Numeric pad. Programmable reversible video; aux. port; self-test mode; protect mode; block mode; tabbing; addressable cursor. Microprocessor controlled; programmable underline; line and character insert/delete.

OUR PRICE ONLY $789

INTERTUBE II by Intertec

12" Display; 24 x 80 format; 18-key numeric keypad; 128 upper/lower case ASCII characters; Reverse video, blinking; Complete cursor addressing and control; Special user-defined control function keys; Protected and unprotected fields; Line insert/delete and character insert/delete editing; Eleven special line drawing symbols.

Only $799

BANTAM 550
From PERKIN ELMER

NOW FROM US AT

$799

Add $20 for anti-glare CRT

HAZELTINE

HAZELTINE 1500
List $1225 Only $989

1400, List $850 . . . . $ 699
1410 w/numeric keypad, List $900 . . . 749
1420 w/lower case & numeric pad 849
1500, Kit, List $1125 799
1510, List $1395 1089
1520, List $1650 1389

Check our price and delivery on the new HAZELTINE 1420 with lower case, numeric keypad, and 2-year warranty.

SOROC

120, List $995 . . . . $ 795
IQ-140, List $1495 1249

CENTRONICS PRINTERS

NEW 730 w/friction & tractor . . . . $ 895
779-1 (TRS-80 Line Prtr. 1), List $1245 949
779-2 with tractor, List $1350 1049
702-2 120 cps, bi-directional, tractor, VFU 1995
703-2 185 cps, bi-directional, tractor, VFU 2395
780-2 w/tractor, List $1725 1295

ANADEX
80-COL. DOT MATRIX PRINTER

Complete upper and lower case ASCII char. set, bi-directional at 84 lines/min. Features RS232 20/60 mil current loop and Centronix parallel interface. Ideal for use with TRS-80, Sorcerer, Cromemco, and North Star systems.

OUR PRICE ONLY $895

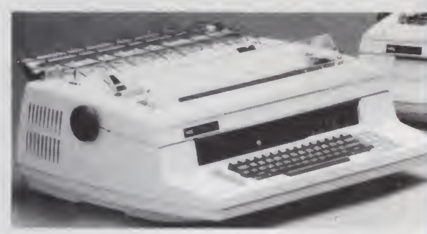
T.I. 810

Bi-directional; 150 cps; logic seeking; adjust able tractor. Available with lower case com pressed print; forms length control or vertica forms control option. Centronics style paralle interface also available.

TI-810 Basic Unit, List $1895 ONLY $1695
TI-810 w/full ASCII (lower case), vertical forms control, and compressed print . $1895

PAPER TIGER®

IDS-440 Paper Tiger, List $995 . . . . $ 895
w/graphics option, incl. buffer, $1194 989
IP-125 w/1210 option*, List $838 . NOW 72
IP-225 w/1210, 1250 options, List $984 83
IP-225 w/tractor, 1210*, 2K buffer, & graphics options, List $1098 89
TRS-80 cable 4
*1210 option is expanded/compressed print.



NEC SPINWRITER™

The Fantastic Letter-Quality Printer at 55 cps
RO's as low as $2475

All prices subject to change and all offers subject to withdrawal without notice. Prices in this ad are for prepaid orders. Slightly higher prices prevail for other-than-prepaid orders, i.e., C.O.D., credit card, etc.

— WRITE FOR FREE CATALOG —

VISA master charge THE INTERBANK CARD

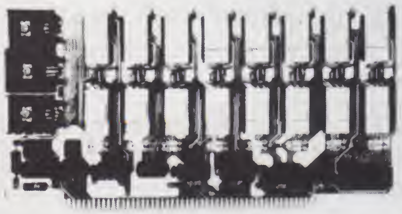
✓ 50

MiniMicroMart, Inc.

1618 James Street, Syracuse NY 13203 (315) 422-4467 TWX 710-541-0431

DIGITAL RESEARCH COMPUTERS
(214) 494-1505

16K EPROM CARD-S 100 BUSS



$59.95 KIT

FIRST TIME OFFERED! BLANK PC BOARD - $28

USES 2708's!

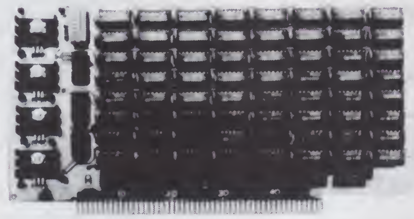
Thousands of personal and business systems around the world use this board with complete satisfaction. Puts 16K of software on line at **ALL TIMES!** Kit features a top quality soldermasked and silk-screened PC board and first run parts and sockets. Any number of EPROM locations may be disabled to avoid any memory conflicts. Fully buffered and has WAIT STATE capabilities.

OUR 450 NS 2708'S ARE $8.95 EA. WITH PURCHASE OF KIT

ASSEMBLED AND FULLY TESTED ADD $30

8K LOW POWER RAM KIT-S 100 BUSS

SALE



PRICE CUT!

$119.50 KIT

21L02 (450 NS RAMS!)

Thousands of computer systems rely on this rugged, work horse, RAM board. Designed for error-free, NO HASSLE, systems use.

ASSEMBLED AND FULLY BURNED IN ADD $35

ALL ASSEMBLED BOARDS ARE TESTED AT 4MHZ.

Blank PC Board w/Documentation **$29.95**

Low Profile Socket Set **13.50**

Support IC's (TTL & Regulators) **$9.75**

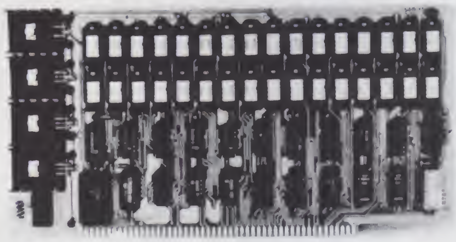
Bypass CAP's (Disc & Tantalums) **$4.50**

16K STATIC RAM KIT-S 100 BUSS

PRICE CUT!

$249 KIT

FOR 4MHZ ADD $25



KIT FEATURES:

1. Addressable as four separate 4K Blocks
2. ON BOARD BANK SELECT circuitry (Cromemco Standard!) Allows up to 512K on line!
3. Uses 2114 (450NS) 4K Static Rams
4. ON BOARD SELECTABLE WAIT STATES
5. Double sided PC Board, with solder mask and silk screened layout. Gold plated contact fingers
6. All address and data lines fully buffered
7. Kit includes ALL parts and sockets
8. PHANTOM is jumpered to PIN 67
9. LOW POWER under 1.5 amps TYPICAL from the +8 Volt Buss
10. Blank PC Board can be populated as any multiple of 4K

BLANK PC BOARD W/DATA-**$33**

LOW PROFILE SOCKET SET-**$12**

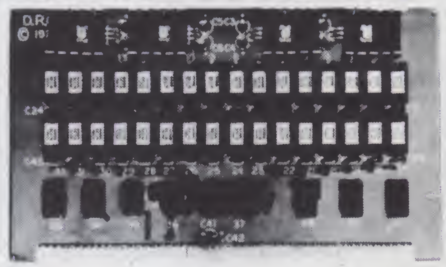
SUPPORT IC'S & CAPS-**$19.95**

ASSEMBLED & TESTED-**ADD $30**

OUR #1 SELLING RAM BOARD!

16K STATIC RAM SS-50 BUSS

PRICE CUT!



$239 KIT

FULLY STATIC AT DYNAMIC PRICES

FOR SWTPC 6800 BUSS!

ASSEMBLED AND TESTED - $35

KIT FEATURES:

1. Addressable on 16K Boundaries
2. Uses 2114 Static Ram
3. Runs at Full Speed
4. Double sided PC Board. Solder mask and silk screened layout. Gold fingers
5. All Parts and Sockets included
6. Low Power: Under 1.5 Amps Typical

BLANK PC BOARD—**$26** COMPLETE SOCKET SET—**$12**

SUPPORT IC'S AND CAPS—**$19.95**

NEW! **STEREO!** NEW!

S-100 SOUND COMPUTER BOARD

At last, an S-100 Board that unleashes the full power of two unbelievable General Instruments AY3-8910 NMOS computer sound IC's. Allows you under total computer control to generate an infinite number of special sound effects for games or any other program. Sounds can be called in BASIC, ASSEMBLY LANGUAGE, etc.

COMPLETE KIT! $84.95 (WITH DATA MANUAL)

BLANK PC BOARD W/DATA $31

KIT FEATURES:

- TWO GI SOUND COMPUTER IC'S.
- FOUR PARALLEL I/O PORTS ON BOARD
- USES ON BOARD AUDIO AMPS OR YOUR STEREO
- ON BOARD PROTO TYPING AREA.
- ALL SOCKETS, PARTS AND HARDWARE ARE INCLUDED.
- PC BOARD IS SOLDERMASKED, SILK SCREENED, WITH GOLD CONTACTS.
- EASY, QUICK, AND FUN TO BUILD, WITH FULL INSTRUCTIONS
- USES PROGRAMMED I/O FOR MAXIMUM SYSTEM FLEXIBILITY.

Both Basis and Assembly Language Programming examples are included.

SOFTWARE:

SCL Interpreter coming soon! Our new Sound Command Language interpreter along with the Register Examine/Modify (REM) routines and Sound Effects Library (SEL) will be available soon in EPROM. SCL makes sound effects programming generally easier and quicker than that written in Basic or Assembly Language. An SCL users group will be formed, and the best new SCL programs submitted will be added to the Sound Effects Library in EPROM.

S-100 Z80 CPU CARD

Perfect for OEM's

$169.95

WIRED! NOT A KIT!

4 MHZ

ASSEMBLED AND TESTED! READY TO USE! Over 3 years of design efforts were required to produce a **TRUE** S-100 Z80 CPU at a genuinely bargain price!

BRAND NEW!

FEATURES:

- 2 or 4 MHZ Operation
- Generates MWRITE, so no front panel required
- Jump on reset capability
- 8080 Signals emulated for S-100 compatability
- Top Quality PCB. Silk Screened, Solder Masked, Gold Plated Contact Fingers

PRICE CUT! SALE!

LOW POWER - 250NS 2114 RAM SALE!

8 FOR $49.95

4K STATIC RAM'S. MAJOR BRAND, NEW PARTS.

These are the most sought after 2114's, LOW POWER and 250NS FAST

***SPECIAL SALE:* $7.50 ea. or 8 FOR 49.95**

NEW! G.I. COMPUTER SOUND CHIP

AY3-8910. As featured in July, 1979 BYTE! A fantastically powerful Sound & Music Generator. Perfect for use with any 8 Bit Microprocessor. Contains: 3 Tone Channels, Noise Generator, 3 Channels of Amplitude Control, 16 Bit Envelope Period Control, 2-8 Bit Parallel I/O, 3 D to A Converters, plus much more! All in one 40 Pin DIP. Super easy to interface to the S-100 or other busses.

SPECIAL OFFER: $14.95 each Add $3 for 60 page Data Manual.

Digital Research Computers
(OF TEXAS)

P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 494-1505

TERMS: Add $1.00 postage, we pay balance. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCharge. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. 90 Day Money Back Guarantee on all items. Orders over $50, *add 85¢ for insurance.*

NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE SUPPLIERS OF CPM SOFTWARE.

51

ADVANCED COMPUTER PRODUCTS

GET YOUR 1980 CATALOG

FIRST TO OFFER PRIME PRODUCTS TO THE HOBBYIST AT FAIR PRICES!

1. **Proven Quality** Factory tested products only.
2. **Guaranteed Satisfaction**
3. **Over $1,000,000.00 Inventory**

1980 CATALOG NOW AVAILABLE.

Send $2.00 for your copy of the most complete catalog of computer products. A must for the serious computer user.

STATIC RAM BOARDS

Now less than 1 1/2¢ per bit!

S-100 32K (uses 2114)

ASSEMBLED		Kit	
450ns.	499.00	450ns.	469.00
250ns.	539.00	250ns.	499.00

Bare Board 49.95
Bare Board w/all parts less mem. 99.95

S-100 16K (S-100 Compatible) WOW!
- Low Power
- 2 MHz or 4 MHz
- 4K Bank Addressable
- Extended Memory Management
- Assembled & Tested

2 MHz . . . **$250.00**
4 MHz . . . **$265.00**

LOGOS I 8K
ASSEMBLED

450 ns	149.95	KIT 450ns	125.95
250ns	169.95	250ns	149.95

Bare PC Board w/Data $21.95
"Special Offer" Buy (4) 8K 450ns Kits $117.00

TRS-80 ADD-ON DISK
- VISTA V-80 MINIDISK
  - 23% More Storage Capacity · 40 Tracks
  - 40 track patch free
- New Double Density Expansion Module $239.00
  - 2 Drive Cable Add $29.95
  - 4 Drive Cable Add $39.95
  - Dual Drive in Single Cabinet $770.00

395.00

S-100 ADD-ON MINI-DISK
- VISTA V-200 Minifloppy Add-on System/Horizon, S-100, etc.
- 204K Byte Capacity
- Double Density Drive
- CP/M & Basic "E"
- Double Density Controller
- Case and Power Supply
- Assembled & Tested Ready to Plug In
- New Dual Exidy Add-on only $1199.00

V-200 699.00

S-100 ADD-ON DISK SUBSYSTEM
- VISTA V-1000 Floppy Disk Subsystem
- (2) Shugart 8" Disk Drives
- Case with Fan & Power Supply
- VISTA Floppy Disk Controller Card
- CP/M Disk Operating System
- Fully Factory Assembled & Tested

V-1000 1699.00
DOUBLE DENSITY $1999.00

LOW-COST ADD-ON DISK SUBSYSTEM KIT
- (2) 8" Drives
- VISTA Floppy Controller Card
- Power Supply and Interface Cable
- CP/M Disk Operating System
- Box of 10 Diskettes

Low Cost Sys. 1299.00

IMS STATIC RAM BOARDS
- Memory Mapping
- Low Power
- Phantom
- Assembled & tested

$699.00

Recommended by Alphamicrosystems

	250 ns.	450 ns.
8K Static	$209.00	$189.00
16K Static	$449.00	$399.00
32K Static	$729.00	$629.00

ANADEX PRINTER NEW APPLE VERSION
Model DP-8000 compact, impact, parallel or serial. Sprocket feed, 80 cols, 84 lines/min., bi-directional.
New only $875.00
DP-8000AP (for Apple) $875.00

SD SYSTEMS BOARDS

	KIT	ASSEM
SBC 100 Single Board Computer (2MHz)	$265.00	$ 349.00
SBC 200 Single Board Computer (4MHz)	299.00	399.00
Z80 Starter System	319.00	449.00
VDB 8024 Video Display Board	335.00	459.00
Versa-Floppy II	325.00	429.00
Expando PROM	115.00	225.00
SD100 Computer System w/64K	–	6995.00
SD200 Computer System w/64K	–	7995.00

MOTOROLA EXORCISER COMPATIBLE

9600 MPU Module w/6802 CPU	$495.00
9601 16 Slot Mother Board	175.00
9602 Card Cage (19" Retma Rack Mount)	75.00
9603 8 Slot Mother Board	100.00
9604 Switchmode System Power Supply	250.00
9610 Utility Prototyping Board	36.00
9616 Quad 8K Eprom Module	CALL
9620 16 Channel Parallel I/O Module	295.00
9622 Serial/Parallel I/O Combo	CALL
9626 8K Static RAM Module	295.00
9627 16K Static 450ns	495.00
9630 Card Extender	68.00
9640 Multiple Programmable Timer (24 Timers)	395.00
9650 8 Channel Duplex Serial I/O	395.00
96103 32/32 I/O Module	275.00
96702 32 Point Reed Relay Module	350.00

6800 BARE BOARDS

9620-0	$45.00	9603-0	27.00
9626-0	45.00	9600	55.00
9650-0	45.00	96103	55.00
9601-0	50.00	96702	55.00

Also AMI EVK System in Stock

APPLE/EXIDY/EXPANDO TRS 80 16K-UPGRADE KIT
- 16K with Jumpers & Instructions for either Level I or Level II . . . $62.95
- 16K for Apple II Upgrade . . . $62.95
- Special: TRS80 Schematic . . . $ 4.95
- Expansion Interface Schematic . . $ 4.95
- Expansion Interface Connector . . . 7.95

EXPANDORAM II MEMORY KITS
- Bank Selectable
- Uses 4116 200 ns.
- Write Protect
- Power 8VDC, ±16VDC
- Phantom
- Up to 4 MHz

Expando 64 Kit (4116)

16K	$290.00	48K	$444.00
32K	$365.00	64K	$519.00

INDUSTRIAL SELECTRIC PRINTER
For TRS-80 . . . . $1299.00
Serial I/O 1299.00
Refinished, with warranty.

TASA touch activated solid state KEYBOARD
- Full 128 position 8-bit ASCII
- 3 color code, 55 positions
- Immune to static charge
- Low power

Only . . . . . 79.95 Optional stand 15.00

UV "Eprom" Eraser
Model UVs-11E $69.95
Holds 4 Eprom's at a time. Backed by 45 years experience.
Model S-52T . . . $265.00
Professional Industrial Model

TARBELL FLOPPY INTERFACE
★ Z80/8080 S 100 Compatible ★ Uses CPM

Double Density	Kit 310.00
Assembled	415.00
Standard Floppy Controller	Kit 179.95
Assembled (Specify Floppy)	259.00

Note: For CPM Add $70.00 Documentation Add $20.00
Vista Double Density 5¼" Controller Assem. $299.00
SD Versa Floppy Kit II Assembled $325/$429
Tarbel Cassette I/O Kit $115.00
Sale ★ 1771-01 Floppy Chip $24.95 ★ 1791-01 $36.95
VISTA Floppy Controller 8" (Shugart Comp.) $259.95

NEW CENTRONICS 730-1 PRINTER
WOW $895.00
Uses any paper roll, fanfold, single sheets, 96 character ASCII, 7 x 7 dot matrix, 50 CPS, RS232 or parallel I/O

Z-80/Z-80A/8080 CPU BOARD
- On board 2708
- 2708 included (450ns.)
- Power on jump
- completely socketed
- Z-80 Assembled and Tested $185.00
- Z-80 Kit $129.95
- Z-80 Bare PC Board . . . . $ 34.95
- For 4MHz Speed Add $15.00

8080A Kit . . . . $ 99.95
8080A Assembled $149.95

S-100 MOTHERBOARD SPECIAL
8 slot expandable w/9 conn.
reg $69.95 . . . . NOW $52.95

SIEMEN'S FLOPPY SALE
- Special buy while supply lasts.
- 8" Drive with Double-Density
- 90 Day Warranty

SIEMENS $429.00
SHUGART 801R $475.00

ACOUSTIC MODEM
NOVATION CAT™
0-300 Baud
Bell 103
Answer, Originate $179.95

ACOUSTIC COUPLER SPECIAL
AJ MODEL A30
SPECIAL PURCHASE
OF SURPLUS UNITS
AVAILABILITY LIMITED $29.95

DATA BOOKS • COMPUTER BOOKS

1980 IC Master	59.95	Intel MCS 80 Manual	7.95
NSC TTL Data	3.95	Intel MCS 40 Manual	4.95
NSC Linear	4.95	AMD 8080A Manual	5.95
NSC Linear App Notes II	3.95	AMD Schottky Databook	4.95
NSC CMOS	3.95	AMI MOS/LSI Data	3.95
NSC Memory	3.95	GI MOS/LSI Data	4.95
Intel Databook	4.95	Harris Analog Databook	4.95
Intel MCS 85 Manual	7.50	TI Linear Control Data	3.95

SALE • OSBORNE BOOKS • SALE

	Reg.	Sale
Intro to Micros Vol 0	~~8.50~~	7.75
Intro to Micros Vol I	~~8.50~~	7.75
8080A Programming	~~8.50~~	7.75
6800 Programming	~~8.50~~	7.75
Z80 Programming	~~8.50~~	7.75
Vol II Some Real Microprocessors w/Binder	~~30.00~~	27.50
Vol III Some Real Support Devices w/Binder	~~20.00~~	18.50
Intro to Micros Vol III	~~20.00~~	18.50

SALE • DILITHIUM COMPUTER BOOKS • SALE

Understanding Computers	~~8.95~~	7.95
8080Microcomputer Experiments	~~13.95~~	11.95
Beginning BASIC	~~9.95~~	8.95
Beginners Glossary & Guide	~~6.95~~	5.95
Peanut Butter & Jelly Guide to Computers	~~7.95~~	8.95
8080 Machine Language Programming	~~7.95~~	6.95
Home Computers Vol I Hardware	~~7.95~~	6.95
Home Computers Vol II Software	~~13.95~~	11.95
Starship Simulator	~~7.95~~	6.95

MICROPROCESSORS

Z8001 16 bit to 8Mb	$189.00
Z8002 16 bit to 64K	149.00
Z80	**10.75**
Z80A	14.50
F-8 (3850)	16.95
2650	16.95
CD1802	13.95
8080A	8.50
8080A-4MHz	19.95
SALE 8085	**19.95**
8008-1	14.95
2901	9.90
2901A	14.95
2903 4-bit Superslice	29.95
TMS 9900JL	49.95
CP1600	39.95
6502	11.50
6502A	16.95
IM6100	29.95
6800	**11.75**
6800B 2.0 MHz	19.95
6802P	17.95
8035	19.95
8755	49.95
8748	69.95
6809	64.95
8086	69.95

ADVANCED SUPPORT

AM9511 Arith. Processor	175.00
9512 Arith Processor	175.00
9513 Univ Timing	79.95
AM9517 DMA Controller	18.95
AM9519 Universal Interrupt	18.95

Z-80 SUPPORT CHIPS

Z80-PIO	2.5 MHz	8.75
Z80A-PIO	4.0 MHz	12.95
Z80-CTC	2.5 MHz	8.75
Z80A-CTC	4.0 MHz	12.95
Z80-DMA	2.5 MHz	29.95
Z80A-DMA	4.0 MHz	36.95
Z80-SIO/0	2.5 MHz	35.95
Z80A-SIO/0	4.0 MHZ	39.40
Z80-SIO/1	2.5 MHz	35.95
Z80A-SIO/1	4.0 MHz	39.40
Z80-SIO/2	2.5 MHz	35.95
Z80-SIO/2	4.0 MHz	39.40

8080/8085 SUPPORT

8155/8156 I/O	24.95
8755 I/O with Eprom	64.95
8202 Dyn Ram Cont	34.95
8205/74S 138 Decoder	3.95
8212 8 bit I/o	2.75
8214 Priority Int	5.25
8216 Bus Driver	2.75
8224 Clock Gen	2.95
8224-4 (4MHz)	9.75
8226 Bus Driver	2.75
8T26 Bus Driver	2.95
8228 Sys Control	5.50
8238 Sys Cont	5.50
8251 Prog I/O	6.95
8253 Int Timer	16.95
8255 Prog I/O	9.90
8257 Prog DMA	16.95
8259 Prog Int	17.95
8275 CRT Controller	59.95
8279 Prog Keyboard	18.95

6800 SUPPORT CHIPS

6810 128 x 8 Ram	4.75
6820 PIA	5.95
6821 PIA	6.50
6828 Priority Int	9.95
6834-1 512 x 8 Eprom	16.95
6845/HD46505 CRT Cont	39.95
6847 Color CRT	49.95
6850 ACIA	5.95
6852 Serial Adapter	5.95
6860 Modem	10.95
6862 Modulator	11.95
6871A 1.0MHz OSC	25.95
6875	8.25
6880 Bus Driver	2.95
MC68488	19.95
68047	24.95

1802 SUPPORT CHIPS

1821 SCD 2K RAM	25.00
1822 SCD 256 x 4 RAM	16.95
1824 CD 32 x 8 RAM	9.95
1852 CD 8 bit I/o	10.95
1854 Uart	10.95
1856 CD I/O	8.95
1857 CD I/O	8.95
1861	12.95

6502 SUPPORT CHIPS

6520 PIA	7.50
6522 Mult	11.95
6530-002.003 004.005	18.95
6532	19.95
6551	19.95

PROMS

2708 450 ns	8.25
2708-6 650 ns	7.50
1702A	4.95
2732	74.95
2716-5V	29.95
2716-5V 12V	29.95
2758-5V	29.95
5203AQ	13.95
5204AO	14.95
IM 5610	3.90
SALE 8223 32 x 8	**2.95**
82S115 512 x 8 (TS)	16.95
82S123 32 x 8	4.90
82S126 256 x 4	4.90
82S129 256 x 4 (TS)	4.90
82S130 512 x 4 (OC)	6.50
82S23	6.95
82S131	14.95
82S137	14.95

NOTE: WE PROGRAM PROMS

CHARACTER GEN.

2513-001 (5V) Upper	9.50
2513-005 (5V) Lower	10.95
2513-ADM3 (5V) Lower	14.95
MCM6571	11.75
MCM6571A	11.75
MCM6574	14.50
MCM6575	14.50

UARTS/BAUD RATE

TR1602B (5V 12V)	3.95
AY51013 (5V 12V)	4.95
AY51014A/1612 (5-14V)	6.95
AY51015A/1863 (5V)	6.95
TMS 6011 (5V 12V)	5.50
IM6402	7.95
IM6403	8.95
2350 USRT	9.95
1671B Astros	24.95
SALE TR1472B	**9.95**
MC14411	11.95
4702	14.95
WD1941	9.95
COM 5016	16.95

KEYBOARD ENCODERS

AY5-2376	13.75
AY5-3600	13.75
HD0165	9.95
74C922	9.95
74C923	9.95

STATIC RAMS

	1-24	25-99	100
21L02 450ns	1.30	1.25	.99
21L02 250ns	1.59	1.55	1.15
2111	3.75	3.65	3.55
2112-1	2.95	2.85	2.65
2101-1	2.90	2.70	2.55
2114L-250ns (4045)	8.95	7.50	6.50
2114L-450ns (4045)	5.50	5.25	**4.75**
4044 250 ns	8.95	7.50	6.90
4044 450ns	5.95	5.95	**4.75**
EMM4200A	9.75	8.75	7.95
EMM4402	7.95	7.25	6.25
EMM 4804	12.50	11.50	9.95
5101C-E	7.95	7.95	7.25
AMD9140/41	10.95	10.25	9.25
AMD9130/31	2.95	11.95	10.25
1101	1.95	1.75	1.25
P2125/93425 (45ns)	9.95	8.35	8.25
6508 1K x 1 CMOS	7.95	7.95	7.25
6518 1K x 1 CMOS	7.95	7.95	7.25
74S189 64 bit Ram	3.95	3.25	2.50
2147 Low Power 4K Static	19.95	18.95	16.95

DYNAMIC RAMS

416/4116 16K (16 Pin)	8.75
Set of 8 416's	62.95
4115 8K (16 Pin)	6.95
4050 4K x 1 (18 Pin)	4.95
4060 4K x 1 (22 Pin)	4.95
4096 4K x 1 (16 Pin)	3.95
2104 4K x 1 (16 Pin)	4.75
4027 4K x 1 (16 Pin)	4.95

5261	1.95	1103	1.95
5262	1.95	4008L	4.95
5270	4.95	6605	7.95
5280	4.95	6604	4.95
5290	12.45	6002	1.50

64K RAMS $175.00

CHARGE COUPLED DEVICES
16K CCD First time offered Fairchild 460 CCD 16K Memory (now you can experiment with CCD technology at a reasonable price. 17 page Application note supplied with each order. Quantity limited)

$18.95 each (reg. 43.00)

SOCKETS

# Pins	Lo-Pro Soldertail	Wire Wrap 3 Level Tin	Gold
8	15	32	45
14	19	36	59
16	20	38	62
18	24	59	84
20	29	69	99
22	24	79	1.10
24	38	85	1.20
28	43	1.10	1.49
36	58	1.25	1.69
40	60	1.40	1.89

ZERO INSERTION FORCE
16 Pin $5.50 24 Pin $7.50 40 Pin $10.25

DISPLAYS/OPTO/LEDS

★ 7 SEGMENT ★ CALC ★ CLOCKS ★

DL 704 (CC) DL 707 (CA) .300 Red	.99
FND 357 (CC) .357 Red	.99
FND 500/503 (CC) .500 Red	.99
FND 507/510 (CA) .500 Red	.99
FND 800/803 (CC) .800 Red	1.75
FND 807/810 (CA) .800 Red	1.75
XAN 3062 .500 Green	1.15
HP5082-7731 (CA) .300 Red	.99
9 Digit Bubble Mini Calc Display	.99
9 Digit Panaplex Display .400	.99
9 Digit Fluorescent .300	.99
MA1002A LED 12 hr Clock Module	10.95

★ HEX DISPLAYS ★ ENCODED DISPLAYS ★

HP 5082-7340 Red Hexidecimal	15.95
HP 5082-7300 Red Nymeric	14.95
TIL 305	6.95
TIL 308 Number w/Logic	8.95
TIL 309 Number w/Logic	8.95
TIL 311 Hexadecimal	12.95
MAN 2A 320 Red Alpha-Numeric	[illegible]
MAN 10A 270 Red Alpha-Numeric	8.95

IC SPECIAL PURCHASE

LF 13508 JFET Analog Multi 8 bit	8.95
ICM 7045 Precision StopWatch	23.95
ICM 7205 CMOS LED Stopwatch/Timer	19.95
ICM 7207 Oscillator Controller	7.50
ICM 7208 Seven Decade Counter	18.95
ICM 7209 Clock Generator	6.95
ICL 7107 3½ Digit A/D (LED)	14.95
MC14433P 3½ Digit A/D Converter	13.95
ICL 8211 Voltage Reference	1.95
LM 390N Battery Op Audio Amp	3/1.00
LM 1800N PLL FM Sterero	3/1.00
LM 1820N AM Radio	3/1.00
LM 1850N Ground Fault IC	3/1.00
LM 2900N Ouad Amplifier	2/1.00
LM 2901N (HM) Ouad Comparator	2/1.00
LM 2917N Freq to Volt Conv	2/1.99
AY-3-3550 4¾ digit DMM	24.95
AY-5-3507 40MHz DVM	9.95
MEM 4963 Mostet Smoke Detector	11.95
27S08 32 x 8 Prom (IM 5600)	2.95
G115M 6 channel Mosfet	1.49
ULN 2003A 7 channel Driver	.99
10146 1K x 1 ECL Ram	9.95
MCM 14505 64 bit Static Ram	6.95

7520/25	.50	LM 377	1.25
1488/1489	2/1.99	LM 387	1.25
MK5014 Calc	3/1.99	LM 3302	3/1.99
74S89	3/1.99	LM 4558	3/1.99
78H05K	6.95	RC 4136	2/1.99
LM323K	4.95	RC 4131	2/1.99
MC1372	8.95	COM 5027	39.95
		COM 5037	49.95

PROCESSOR TECHNOLOGY
NKRA Dynamic Memory Boards, Assembled Without RAM, PROM, or Data Delay Line
Up to 64K per Board $139.95

FLOPPY DISKS
CERTIFIED! 40 TRACKS DOUBLE DENSITY

8" 3740 IBM Compat	$4.39
5¼" Hard 10 Sector	$3.99
5¼" Soft Sector 40 Track	$3.99

Vista computer diskettes by Verbatim Minimum order 10 For quantity purchases please call

ALSO AVAILABLE
DOUBLE SIDED DOUBLE DENSITY

PAK OF 10 5¼"	$56.40
8"	$63.90
Floppy disk hard holes and applicators	CALL

FLOPPY DISK I/O

1771-01 8" & Minifloppy	24.95
uPd372 Nec Floppy	49.95
1781 Dual Floppy	29.95
1791 Dual Floppy	36.95
uPd 765 Floppy	49.95

A/D CONVERTERS

8700 8 bit Binary	13.50
8701 10 bit Binary	22.00
8703 8 bit TS	13.50
9400 Volt to Freq Conv	7.25
8750 3½ Digit BCD	13.95
1408L6 6 bit	3.95
1408L8 8 bit	5.95
DAC01 D to A	5.95

TV CHIPS/SOUND

AY38500-1 6 Games B W	4.95
AY38515 Color Converter	2.95
AY38603-1 Roadrace Game	8.95
AY38605-1 Warfare Game	CALL
AY38606-1 Wipeout Game	9.50
AY38607-1 Shooting Gallery	8.95
AY38910 Sound Generator	12.95
SN76477 TI Sound Generator	1.95
MM5320 21 TV Synch Gen	9.95
MM5369 Prescaler	3.95
LM1889 RF Modulator	3.95
MM57100 NSC Color TV	6.95
MM57104 Clock Gen	[illegible]
RF Modulator w Audio	8.95
M&R Modulator	29.95

WAVEFORM GEN.

8038 Function Gen	[illegible]
MC4024VCO	[illegible]
LM566 VCO	[illegible]
XR2206 Function Generator	[illegible]

SHIFT REGISTERS

MM500H Dual25	.50
MM5056N Dual 256	2.95
MM5060N Dual 128	2.95
2510A Dual 100	1.95
2847 Quad 80	4.95
3341 Dual 80	4.95
3351 40 x 9 FIFO	17.95
3357 Ouad 80	6.95
9403 16 x 4 FIFO	24.95
9408 10 Bit Sequencer	9.95

CTS DIPSWITCHES

CTS206-2	1.75	CTS206-7	[illegible]
CTS206-4	1.75	CTS206-8	1.95
CTS206-5	1.75	CTS206-9	1.95
CTS206-6	1.75	CTS206-10	1.95

CONNECTORS (GOLD)

DB25P (RS232)	3.25
DB25S Female	1.75
Hood	1.25
Set w/Hood Sale	$7.50
22/44 W/W S/T KIM	2.95
43/86 W/W S/T MOT	6.50
50/100 S-100 Connector w/w	4.95
50/100 S-100 Connector s/t	3.95

NAKED PC

Z-80 CPU (Ithaca)	$14.95
8080A CPU	34.95
8K Static RAM (Logos)	21.95
16K Static RAM (2114)	29.95
32K Static RAM (2114)	49.95
Floppy I/O (Tarbell)	19.95
Cassette I/O (Tarbell)	29.95
8K Eprom (2708)	[illegible]
1702 Eprom Board	[illegible]
2708/2716 Eprom/Prog	[illegible]
2708/2716 Eprom/WMC	[illegible]
Realtime Clock	[illegible]
ACP Proto Bd (IMSAI)	[illegible]
Vector 8800 Proto	22.20
Vector 8801 MB	[illegible]
ACP Extender w Conn	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
11 Slot Mother Board	[illegible]
6 Slot Mother Board	[illegible]
[illegible]	[illegible]
Proc Tech Bare Boards	CALL

COMPUTER SPECIALS

	LIST	SALE
Apple II Plus w/16K	1195	990
PET 2001-16N	995	895
Exidy Sorcerer w/16K		1099
Compucolor II w/8K	1595	1495
Cromemco Sys III	6990	6290
Horizon I w/32K		CALL
TEI PT208 w/32K dual floppy & CRT	4995	2495
Pascal Microengine	2995	2395
IPSI 1620 Diablo RO	3295	2695
Anadex DP 8000	995	875
Centronics Micro P-1	595	395
Centronics Micro S-1	595	525
Soroc IQ 120	995	795
Teletype Model 43	1349	1150
HiPlot Plotter	1085	899
HiPlot Digitizer	795	735
Intertube II	895	785
SOL 20		1095
Atari 800	999	899
Atari 400	549	499
TI 99/4	1150	1099
Leedex 12" Monitor	149	129
Centronics 730-1	995	895
Trencom T-100	375	349
Trencom T-200	595	525
Sanyo 9" Monitor	220	169
Sanyo 15" Monitor	295	279
Motorola 22 MHz Monitor	296	219

SOCKET SPECIALS

14 Pin w/w-2L	$ .35
16 Pin w/w-2L	.40
18 Pin w/w-2L	.55
20 Pin w/w-2L	.75
40 Pin w/w-2L	1.45
18 Pin s/t	.15
22 Pin s/t	.15

Call for volume pricing

TERMS: Use check M/C VISA, AMEX, CB or COD COD requires 25% deposit Charge Orders please include expiration date Foreign pay U S Funds Order by phone mail or TWX MINIMUM $10.00 Please include magazine issue and phone no SHIPPING U S A add $2.00 for first 2 lbs For surface add 30¢ for additional lbs For air add 70¢ for additional lbs FOREIGN Add 10% shipping and handling COD's $1.85 extra Not responsible for typos Some items subject to prior sale We reserve right to limit quantities some items subject to precious metal adder Retail pricing may vary from Mail Order

RETAIL STORES OPEN MON-SAT

STORE #1 1310 "B" E. Edinger Santa Ana, CA 92705 Showrooms, Retail, Warehouse

STORE #2 674 El Camino Real Tustin, CA 92680 Specializing in Systems

P.O. Box 17329 Irvine, Calif. 92713
Direct Order Lines: (714) 558-8813
(800) 854-8230

FOR INTERNATIONAL ORDERS:
1401 S. Borchard (714) 953-0604
Santa Ana, CA92705 TWX: 910-595-1565

LETTERS

(from page 20)

to use the bad half of the memory chips. I did it and found all my 4115s were fully functional 4116s. Try it—you only need to change one jumper. If nothing else, it will be a good workout for your memory-tester program, and, who knows, you may really have double the memory you thought you did.

Ken Hensey
Sonoma CA

To OSI Owners and Users

I believe there is a bug in OSI's version of ROM BASIC. The following program will run until memory is filled and then bomb.

```
10 A$="THIS IS A LONG STRING"
20 B$="THIS IS ANOTHER EVEN LONGER STRING"
30 C$(1)=A$+B$
40 PRINT C$(1)
50 GO TO 30
```

I don't know if OSI has any plans to reissue ROM's as Commodore has done for PET.

Harold Green
Dearborn MI

Microcomputing in Indonesia

Eight years ago I worked with a Texas Instruments 827A computer with 4K memory. This was my first experience with machine language. In 1975 I was assigned to work with 512K IBM 370/145 systems. I learned COBOL, FORTRAN and PL/I.

In the October 1977 issue of *Popular Science*, I saw an advertisement for the PET-2001. I wanted to buy one, but couldn't afford it. After studying and comparing the PET and TRS-80 for almost two years, I completed a part-time software project, which enabled me to buy a 16K PET.

Since December 1978 I have been working for our Foundation, which maintains a 16K PET and a Level II TRS-80. We receive donations and provide training in BASIC, COBOL, PL/I and FORTRAN, but we need funds for maintaining our computers and to develop our organization, which is nonprofit and taxfree.

You may be surprised to know that a PET 2001/16K costs $2400 (U.S. currency) here—same for a 16K TRS-80. That's why we do not have a printer for our two computers and ask for a donation of a used but usable printer that can be connected to our computer *without* additional interfaces. We have only a standard system here: VDU, keyboard and cassette recorder. If someone or any company will donate us a printer, we will try to pay the freight from the U.S. to here.

We thank you.

Maruto Kolopaking
Informatika Foundation
Box 284
Bogor Indonesia

CLUB NOTES

Nassau Bay TX

The NASA/Bay Area TRS-80 Users' Group meets the second Tuesday of each month at 7:30 PM in the Lockheed Bldg. L-X1, room 2012, Space Park Dr., Nassau Bay TX. Contact Ray Cone, (713) 474-3847, or write to the club at PO Box 57116, Webster TX 77598.

Tulsa OK

The Tulsa Computer Society (TCS) meets the last Tuesday of the month at 7:30 PM at the Tulsa Vocational-Technical School seminar center, 3420 South Memorial Drive. Membership is $6 per year and includes the club newsletter. For information write: The Tulsa Computer Society, PO Box 1133, Tulsa OK 74101.

Orange CA

The 300-member North Orange County Microcomputer Club meets the first Sunday of the month at Chapman College in Orange. The club is S-100 oriented (no TRS-80s). Training sessions are offered every other month. For information, write to Tracy Lenocker, PO Box 3616, Orange CA 92665.

El Paso TX

General meetings of the El Paso Computer Club are held the second Saturday of each month at 9 AM in the Electrical Tech. Bldg. behind the Burguss High School, 7800 Edgemere. The Apple users meet the third Tuesday at 7 PM (call Wade 757-0215 for the location), and the S-100 users meet on the second Tuesday of the month at 7 PM (for location call 584-5393). Annual dues are $10 (students $5) and include the club newsletter. For information contact Wade Bolling, 757-0215.

Green Bay WI

Meeting the second Saturday of the month at 9:30 AM at NWTI is Micro, the Green Bay computer club. For more information contact Stuart Mong, 1824 Glenview, Green Bay WI 54303, (414) 494-5862.

Dalton MA

The first Sunday of every month is the meeting date of Computers Anonymous. Further information about the club is available by writing to Computers Anonymous, Box 263, Dalton MA 01226.

Merritt Island FL

Z-80, 8080 and S-100 systems are the specialty of the Space Coast Microcomputer Club. Membership is $5 a year and includes the club's monthly newsletter, the Enterprise. Meetings are the fourth Thursday of every month at 7:30 PM at the Merritt Island Public Library Auditorium. Contact Ray Lockwood, 315 Inlet Ave., Merritt Island FL 32952, (305) 452-2159.

San Antonio TX

For information about the Alamo Computer Enthusiasts contact David Sanson, 4847 Castle Shield, San Antonio TX 78218.

OSI Superboard Club

Owners of OSI, CIP and Superboards can now get updated information in a newsletter to be published bimonthly. Subscribers are encouraged to submit programs, ideas, technical data, hints and suggestions. Send a SASE with $1 for further information to Superboard Club, Box 55, Agincourt, Ontario, Canada M1S 3B4.

Apple 1 Library

Apple 1 owners: the library has the information for updating to the 16K chips. FOCAL, an implementation of Digital Equipment's language, is also ready. Send SASE to Joe Torzewski, 51625 Chestnut Rd., Granger IN 46530.

SYM-1 Users' Group Newsletter

Sym-Physis' first issue, Jan.-Feb., has been getting a good response from its readers. The newsletter is published bimonthly; the rate being $9 per year (USA/Canada). Articles dealing with any aspect of the SYM-1 and its relatives are solicited. For subscription, send a check to SYM Users' Group, PO Box 315, Chico CA 95927.

CALENDAR

Ottawa, Ontario

The Second Annual National Capital Computer Trade Fair will be held at the Ottawa Civic Centre from April 30 to May 2, 1980. There will be over 100 suppliers and manufacturers exhibiting computer, data processing and test equipment, along with computer related supplies and services. This show is oriented to Ottawa's established computer users and first-time buyers. For more information, contact: Laing & Laing Marketing, 145 Bradford St., Ottawa, Ontario Canada K2B 5Y9, (613) 829-6228.

Philadelphia PA

The fifth Produx 2000 will be held May 21–23 at the Philadelphia Civic Center. This is a sales-oriented exposition of business products and personal and business computers. For information, contact Produx 2000, Inc. (215) 457-2300.

Trenton NJ

The fifth annual Trenton Computer Festival will run April 19–20 with a 5-acre outdoor flea market and indoor commercial exhibitor area. There will be 30 speakers, user group sessions and demonstrations. Computer conference sessions and forums will be held and there will be a Saturday night banquet. TCF-80 will be held at Trenton State College, just outside of Trenton NJ. For information, contact Dr. Allen Katz, Trenton State College, Hillwood Lakes, PO Box 940, Trenton NJ 08625. (609) 771-2487.

Clemson SC

The second Clemson Small Computer Conference and Exhibit will be held at Clemson University, Clemson SC on May 21-22, 1980. This program will consist of presentations, discussions and tutorials. There will be displays of equipment from many vendors and manufacturers. Emphasis will be on: business, industry, engineering, science and education. For registration information, contact J. K. Johnson, Continuing Engineering Education, Clemson University, Clemson SC 29631. For information about presentations or equipment exhibits, contact W. J. Barnett, Electrical and Computer Engineering Department, Clemson University, Clemson SC 29631.

Washington D.C.

The sixth annual Federal DP Expo will be held April 28-30 at the newly-expanded Hotel Sheraton Washington. More than 200 companies, vying for their share of the multi-billion dollar U.S. Government computer marketplace, will be exhibiting a broad range of computer-related products, software and services. The current exhibitors roster lists almost 160 companies, including AT&T, Burroughs Corp., McDonnell-Douglas Corp., NCR Corp., Textronix and Wang Laboratories. Admission to the exhibit floor is free to government employees and exhibitor guests, $10 to all others. The first day luncheon will feature a keynote speaker. A buffet lunch will be served on the second and third days.

St. Paul MN

The North Area Repeater Association will sponsor a swapfest and exposition for personal computer hobbyists and radio amateurs on May 31 at the Minnesota State Fairgrounds in St. Paul. Exhibits, booths and prizes. Admission: $3. For information or reservations, write Amateur Fair, PO Box 30054, St. Paul MN 55175.

CONTEST

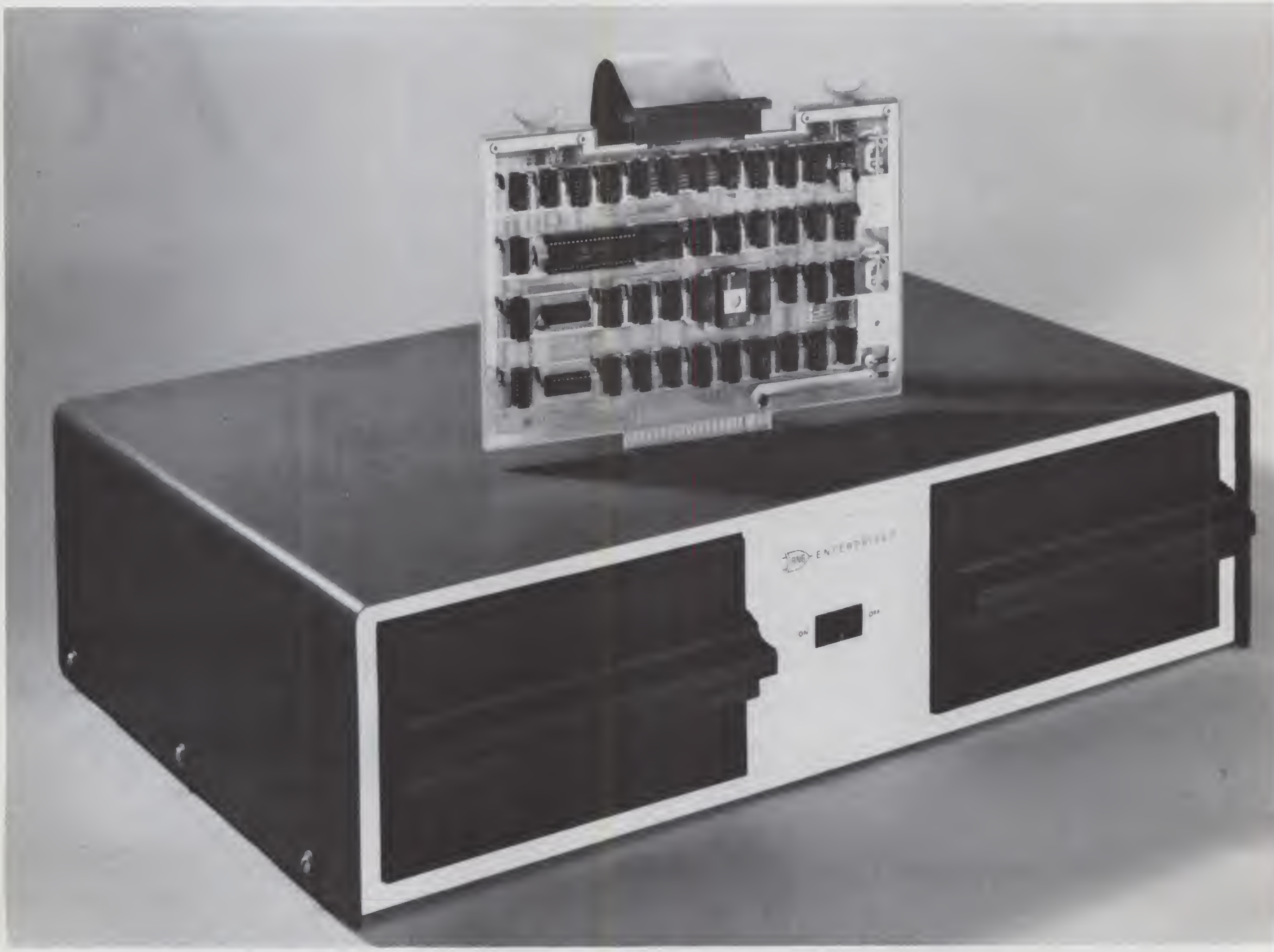
Contest Ends

We're sorry to report that we've discontinued our "best article of the month" contest. The contest ran for two years. Next month we will announce the winner of the "best article of the year" for the contest's second year. To all the winners—again—congratulations. Thanks to all who participated. Maybe we'll do it again sometime.—Editors.

JOB LOT BIDDING

Manufacturers or dealers with job lots of merchandise, systems, software, publications, parts, test equipment, printers, terminals, disks, tapes or monitors should contact Sherry Smythe at (603)924-3873. KB/80/ISI need these for the lab, and we want to bid on your job lot.

VAK-7 FLOPPY



The VAK-7 Disk System was specifically designed for use with AIM-65, SYM-1, and KIM-1 Microcompute Systems. The VAK-7 will plug directly into the VAK-1 Motherboard or with the addition of regulators into the KIM-4* Motherboard. The VAK-7 is a complete full size (8") FLOPPY DISK SYSTEM. This system will READ WRITE, and FORMAT both IBM SINGLE and DUAL DENSITY diskettes. Single-Sided is standard and Dual-Sided is optional. Our Single-Sided drives are set up so they can be converted at a later date to Dual-Sided by the factory, if your storage needs increase.

The VAK-7 system occupies a 4K address space. The system has a 1K block of D.M.A. RAM as a transfer buffer. Also, a 1K block of RAM reserved for D.O.S. pointers, drive status, and catalog information. The remainder o the address is occupied by the resident 2K MINI-DOS. This MINI-DOS is a complete set of subroutines to Read Write, and Format.

DISK SYSTEM

The MINI-DOS is not a high level Disk Operating System, but contains all the elementary subroutines for implementation of a high level DOS. Since all the functions are in subroutines, the implementation of this system into a dedicated system is simplified.

MINI-DOS SUBROUTINES

Block Move	Read/Write Deleted Data
Seek Track	Format Disk/Test For Bad Sectors
Recalibrate Disk	Initialize Disk
Sense Interrupt Status	Physical Copy (Disk to Disk)
Read/Write Data	Self Test

The VAK-7 is an interrupt driven system, which uses the $\overline{\text{IRQ}}$ vector. Since this is an interrupt driven system, your system processor is only used to move data into or out of the 1K of DMA RAM, issue the command, and check status at the end of the disk operation. Your system processor is free to do other functions, during disk operations because the intelligent disk controller will complete the operation without tying up valuable processor time.

The VAK-7 System comes complete with Disk Controller Board, Interconnecting Cable, a Cabinet with Power Supply (for two Disk Drives) and one Disk Drive. The VAK-7 Controller can handle up to Four Drives.

SPECIFICATIONS:

- Completely assembled, tested, and burned in.
- Occupies address $9000-$9FFF for AIM-65, $9000-$9FFF for SYM-1, or $E000-$EFFF for KIM-1.
- IBM Format; Single Density (128 bytes/sector); Dual Density (256, 512, or 1024 bytes/sector).
- All IC's are in sockets.
- Fully buffered address and data bus.
- Standard KIM-4*BUS (both electrical pin-out and card size).
- Designed for use with a regulated power supply, but has provisions for adding regulators for use with an unregulated power supply.
- Dimensions: Board—10" wide x 7" high (including card-edge)
  Cabinet—23.5" wide x 6.5" high x 16" deep.
- Power Requirements: +5V DC @ 2 Amps.
  117V AC 60Hz @ 2 Amps.

**KIM-4 is a product of MOS Technology/C.B.M.*

PRICE:

Single-drive, 1-sided	$1,299.00
Dual-drive, 1-sided	1,898.00
Single-drive, 2-sided	1,499.00
Dual-drive, 2-sided	2,398.00

Plus Shipping	UPS	Mail (APO, FPO)	International
Single Drive	12.00	32.00	Shipped Air Freight. Freight charges collect.
Dual Drive	16.00	44.00	

For Alaska and Hawaii, use mail rates.

We manufacture a complete line of high quality expansion boards. Use reader service card to be added to our mailing list, or U.S. residents send $1.00 (International send $3.00 U.S.) for airmail delivery of our complete catalog.

RNB ENTERPRISES ✓52
INCORPORATED

2967 W. Fairmount Avenue
Phoenix AZ 85017
(602)265-7564

master charge VISA

✓Reader Service—see page 257

Reader Service—see page 257

BUY DIRECT FROM NCE/ CompuMart

Buy Direct from the Largest Commodore Dealer in the Country — NCE/CompuMart.

8K-Keyboard N	$795
16K-Keyboard B	$995
16K-Keyboard N	$995
32K-Keyboard B	$1,295
32K-Keyboard N	$1,295

B — large keyboard (graphics not on keys)
N — large keyboard with graphics symbols

SUPER SAVER. Thanks to Commodore their printer prices have been drastically reduced:

Commodore Printer (tractor feed) .... was $995 **NOW $795**
Commodore Printer (friction feed) .... was $849 **NOW $695**

2 Good reasons for buying your PET™ from CompuMart:

(1) $100 IN FREE ACCESSORIES WITH 16K or 32K PET.

When you buy a 16K or 32K PET, apply $100 toward PET accessories. FREE. Choose from the accessories listed below and indicate on your order that you have reduced the cost of your accessories by $100.

(2) FREE WITH PURCHASE OF ANY PET (Offer good until April 15, 1980)

A Texas Instruments' Hex/Octal Calculator ($60 Value), and a set of 6 PET Workbooks (Worth $36).

PET Accessories

Commodore Dual Floppy Disk Drive	$1,295.00
Second Cassette—from Commodore	$95.00
Commodore PET Service Kit	$30.00
Beeper—Tells when tape is loaded	$24.95
Petunia—Play music from PET	$29.95
Video Buffer—Attach another CRT	$29.95
Combo—Petunia and Video Buffer	$49.95
TNW Bi-Dir.RS-232 printer S-face	$229.00
KIM 1 (A Single Board Computer from Commodore)	$179.00
PET TO IEEE Cable	$39.95
IEEE TO IEEE Cable	$49.95
KIM1 & Power Supply Package Special	$200.00

We have a complete inventory of Apple computers, peripherals, & software in-Stock for immediate delivery — Call us for prices.

$200 in FREE accessories with the purchase of a 48K Apple II reg. or Apple II plus.

16K Apple reg. or plus — $1,195 ($100 in ***free*** acces.)
32K Apple reg. or plus — $1,395 ($150 in ***free*** acces.)
48K Apple reg. or plus — $1,495 ($200 in ***free*** acces.)

Apple Accessories

PASCAL	$495
Micromodem	$379
VisiCalc	$150
The Controller (Business Package)	SALE! $550
The Cashier (POS System)	SALE! $200
Integer BASIC ROM Card	$200
Centronics Printer Interface	$225
Disk and Controller	$595
Parallel Printer Card	$180
Communications Card	$225
Hi-Speed Serial Card	$195
Firmware Card	$200
NEW! AppleWriter Text Editor	$75

NEW FROM EXIDY THE SORCERER II 48K COMPUTER

Z-80 Microprocessor, Full-sized keyboard • ROM PAC Programs • Microsoft BASIC • Powerful Graphics • Serial and Parallel I/O • Dual Cassette I/O Memory Internally Expandable • S-100 Expansion Options.

48K Sorcerer II $1,395

ZENITH

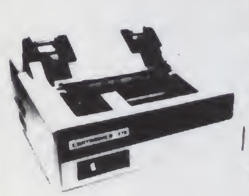
The All-In-One Computer

Save $300 off list on all Zenith Computers, $250 off the printer. Build a great system using both and save $550.

WH89 with 16K RAM	**list $2,295**	**SALE! $1,995**
WH89 with 32K RAM	**list $2,445**	**SALE! $2,145**
WH89 with 48K RAM	**list $2,595**	**SALE! $2,295**
Heath Printer	**list $895**	**SALE! $650**
16K Memory Expansion	**list $150**	**SALE! $110**
Dual-port Serial Interface		**$85**
HD05 Operating System		**$100**
Microsoft Basic		**$100**
WH17 Second Disk Drive		**$550**

Only limited quantities of the Computers and Printers are available. Order Now. (Offer good until April 15, 1980)

CENTRONICS PRINTERS



The 779-2 is a tractor feed printer designed for small business systems. Uses a 5x7 dot matrix to produce all 64 upper case ASCII characters. Centronics is the biggest name in printers and this is their most popular model. List. $1,245 .... SALE! $1,079

The New 730-1 parallel matrix printer is ideally suited for all personal and microcomputer systems. Standard Features include: 100 cps • 80 char./line • 3-way paper handling system • 7x7 dot matrix • 96 character ASCII • List $995 .... SALE!

The Perfect Printer for Small Business Systems

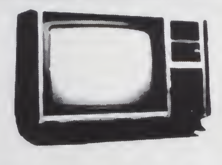
The Paper Tiger Printer From ROAR! Integral Data

Standard features include: 4 character 8.3 to 16.5 cpi • 56 cps at 10 char. per in. • Selectable line spacing • 8 switch-selectable form sizes.

The IDS Graphics Option for the Paper Tiger allows full dot pattern control and includes an expanded 2048-byte buffer (a 256-byte buffer is standard).

IDS Paper Tiger Printer $995
IDS Graphics Paper Tiger Printer $1,094

SPECIAL OFFER EXCLUSIVE FROM COMPUMART!



ZENITH COLOR VIDEO MONITOR

Beautiful Display Capabilities. Excellent for use with Apple, Atari, & Sorcerer Computers. This 13-inch monitor is Zenith's first color video display designed specifically for computers. Features include color and degaussing circuits.

Zenith Color Monitor $499.00

NOVATION CAT™ ACOUSTIC MODEM

- Answer Originate
- Bell 108
- 300 Baud
- Low Profile Design

Looks good, works great! **$179.00**

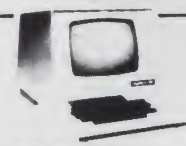
BRAND NEW! The "Star" Modem from Livermore Special Introductory Price $158.00

SANYO MONITORS

9-inch	15-inch
reg. $199	reg. $299
SALE! $169	**SALE! $269**

Perfect for Apple users ...
Sanyo Cassette Recorder **$55**

HAZELTINE TERMINALS AT SPECTACULAR SAVINGS!



Hazeltine 1410-TTY-style keyboard, 8 baud rates from 110 to 9600, 12" screen, 24x80 display, 5x7 dot Matrix Upper Case ASCII character set, 12-key numeric keypad. List Price $895 ...... **SALE! $779**

Hazeltine 1500 — 8 baud rates from 110 to 19,200, and ANSI Standard Keyboard, EIA RS-232, All 94 Upper/Lower Case characters in a high resolution 7x10 matrix display. Much more. List Price $1,145 .... **SALE! $995**

Hazeltine 1520 — Take advantage of this $310 savings while it lasts! The Hazeltine 1520 is a powerful multi-processor terminal. List Price $1,585 .... **SALE! $1,275**

NCE/CompuMart

We've Had a Reputation for Dependability Since 1971.

DEPT. KB40, 270 THIRD ST., CAMBRIDGE, MA. 02142

To Order: 1 (800) 343-5504
In Mass: 1 (617) 491-2700

VISA master charge

Member Computer Dealers Assoc.

REMOTE TERMINAL FOR YOUR PET
Only $98

A self-contained module and program cassette enables your PET to function as a 300 baud terminal. Supports Upper/Lower case. Rubout. Escape and all control functions. Output is TTL. Can be used with THE SOURCE. You will need the Terminal Option, Cat Coupler & EIA Cable to do this. Complete package which includes all 3 available from CompuMart for $279

10-DAY FREE TRIAL

SEND FOR OUR FREE CATALOG

Lear Siegler Terminals & Printers At *Tremendous Savings*

ADM-3A. Industries' favorite dumb terminal for some very smart reasons. 12" diagonal screen • Full or half duplex operation at 11 selectable data rates • 1,920 easy-to-read characters in 24 rows of 80 letters • RS-232C interface extension port • Direct cursor addressing. Reg. Price. $895 .... **Sale! $795**

ADM-31. A terminal that's too smart to be considered dumb. Comes complete with keyboard, control logic, character generator, refresh memory & interface • Displays two pages of text instead of one • Field protect mode • Factory installed selected parameters.
Reg. Price. $1,450 .... **Sale! $1,195**

ADM-42. The semi-intelligent terminal that provides you with flexibility of format, security, editing, interface, and transmission. Two-page display standard (Optionally expandable to eight) • Blanking, blinking, and reverse fields • Three ways to TAB • 16 Function keys do the work of 32 • Detachable keyboard.

ADM-42 with Keyboard. Reg. Price. $1,795 **Sale! $1,595**
ADM-42 without keyboard. Reg. Price. $1,595 .... **Sale! $1,450**

300 Series Ballistic Printers. This application-oriented matrix printer is built to last. Features include: Ballistic Printing • Positive 180 cps • Bi-directional Printing • A character buffer that is optionally expandable to 2,048 characters • 512 character buffer standard.

Model 310 Ballistic Printer (Serial/Parallel)
Reg. Price. $2,045 .... **Sale $1,989**

We Stock Lear Siegler Accessories — Call For Details.

THE SINGLE BOARD DEVELOPMENT SYSTEM ROCKWELL AIM 65

6502 Microprocessor • 20-character, alpha-numeric LED display • Full-size 54-key keyboard with 3 user-defined functions • Fast, on-board 20-column thermal printer • 8K Advanced Interactive Monitor program • Dual cassette interface board • On-board timer • On-board ROM expansion to 12K • 4K on-board RAM • On-board TTY interface • 16 parallel I/O lines • One serial I/O port • KIM-compatible edge connectors for even further memory on I/O expansion.

The CompuMart AIM System combines all of our options for the AIM to give you the capabilities of development systems costing 5 to 10 times as much. This system includes a 4K AIM 65 with BASIC and Assembler, an MTU power supply, a Sanyo tape recorder and an EGI Enclosure for the AIM.

CompuMart AIM System $785.00
4K AIM-65 .... $450.00
Paper for the AIM $4.75

WE'VE GOT IT!

A Calculator, A system, A Whole New Standard.

The finest calculators available for Science, Engineering, & Business.

HEWLETT-PACKARD'S HP-41C

HP-41C

The Calculator.

Features over 130 functions and offers up to 400 lines of program memory or 63 data storage registers — expandable to 319 registers or up to 2,000 lines. RPN Logic. Alpha-numeric capabilities let you communicate with the calculator in English. Customization features allow you to totally reassign the keyboard functions. Continuous memory....... **HP-41C Calculator $288.00**

The System.

Memory Modules. For storing programs or up to 2,000 lines of program memory. **$45.00**
"Extra Smart" Card Reader. Records programs and data back onto blank mag-cards. **$179.00**
The Printer. Upper and Lower case. High resolution plotting. Portable. Thermal operation. **$320.00**
Application Modules.................. **$45.00 EACH.**
Standard pac, Statistics, Math, Financial, & Surveying.

HP SERIES E CALCULATORS

HP-31E — Scientific. Trigonometric, exponential & math functions. **$49.95**
HP-32E — Advanced Scientific with Statistics. All HP-31E functions plus hyperbolas and comprehensive statistics. **$66.95**
HP-33E — Programmable Scientific. A programmable science, math and statistical calculator. **$79.95**
Also available with continuous memory...**$86.40**
HP-37E — Business Management. Best choice for a business/finance calculator........ **$69.95**

HP-38E — Advanced Financial with Programmability. All the features of the HP-37E plus a lot more power. **$108.00**
Also available with continuous memory....... **$144.95**
THE ALL NEW HP-34C. Advanced Continuous Memory Scientific Programmable with an impressive array of programming features. **$144.95**

CompuMart STOCKS THE COMPLETE LINE OF MATROX PRODUCTS. CALL FOR SPECS.

Based on our policy of offering our customers only the finest in microcomputers, CompuMart is pleased to announce that we now carry the new generation of Personal Computers by Atari.™

INTRODUCTORY SPECIALS

To celebrate our commitment to Atari, we are offering the following Atari Specials:
(1) **Buy additional memory for your computer, 8K or 16K, and we will double the amount of memory FREE!**
(A potential savings of $200.)
(2) **Buy the Atari 800 Computer and take $100 off the purchase price of the Atari 810 Disk Drive or the Atari 820 printer.**

ATARI® 800™

PERSONAL COMPUTER SYSTEM

Comes with:
- Computer Console
- BASIC Language Cartridge
- Education System Master Cartridge
- BASIC Language Programming Manual
- 800 Operator's Manual
- ATARI 410 Program Recorder
- Invitation to Programming™ Cassette
- 8K RAM Memory Module™
- 10K ROM Operating System
- Power Supply
- TV Switch Box

SPECIFICATIONS:

High resolution color graphics
57 key full stroke keyboard
Built-in RF modulator for channel 2/3 operation with standard TV set
Composite video output for use with monitor
Internal Speaker
Two cartridge slots for rapid program insertion
Four internal slots for expansion up to 48K RAM
6502B Microprocessor
High speed serial 1/0 port

Atari 800 Computer System $995.95

ATARI® 820™ PRINTER

High resolution dot matrix impact printer
Uses standard ¾ inch roll paper and ribbon
40 characters per line
Speed: 40 characters per second
UL approved

Atari 920 Printer $599.95

ATARI® 810™ DISK DRIVE

Uses standard 5¼ inch diskettes
88K bytes storage per diskette
Up to four disk drive units can operate with the system
Average data access time: 236 milliseconds
Power: AC adapter; UL approved

Atari 810 Disk Drive $699.95

Upgrade your computer with additional memory. (Note that the Atari 800 Computer comes with 8K of RAM memory and will accept up to 48K.)
Atari 8K RAM Memory Module $124.95
Atari 16K RAM Memory Module $199.95

NCE/CompuMart

**Dept. KB40,
270 Third St.,
Cambridge, Mass.
02142** ✓43

**TO ORDER CALL:
1-800-343-5504
IN MASS. CALL: (617) 491-2700**

IMPORTANT ORDERING INFORMATION
All orders must include 4% shipping and handling. Mass. residents add 5% sales tax; Mich. residents 4%, for sales tax.

Phones open from 8:30 a.m. to 5:30 p.m. EST, Mon.-Fri. • P.O.'s accepted from D&B rated companies—shipment contingent upon receipt of signed purchase order • All prices are subject to change without notice • Most items in stock for immediate shipment—call for delivery quotation • In the Ann Arbor area. Our retail store is open 11:00 a.m. to 7:00 p.m. Tues.-Fri., 10:00 a.m. to 5:00 p.m. Saturdays (closed Sun. and Mon.).

***IF NOT SATISFIED RETURN PURCHASE WITHIN 10 DAYS FOR A FULL REFUND.**

✓Reader Service—see page 257

7400 TTL

Part	Price	Part	Price	Part	Price
		SN7470N	.29		
		SN7472N	.29		
SN7400N	.20	SN7473N	.35	SN74160N	.89
SN7401N	.20	SN7474N	.35	SN74161N	.89
SN7402N	.20	SN7475N	.49	SN74162N	1.95
SN7403N	.20	SN7476N	.35	SN74163N	.89
SN7404N	.25	SN7479N	5.00	SN74164N	.89
SN7405N	.20	SN7480N	.50	SN74165N	.89
SN7406N	.29	SN7482N	.99	SN74166N	1.25
SN7407N	.29	SN7483N	.69	SN74167N	1.95
SN7408N	.20	SN7485N	.89	SN74170N	1.59
SN7409N	.20	SN7486N	.35	SN74172N	6.00
SN7410N	.18	SN7489N	1.75	SN74173N	1.25
SN7411N	.25	SN7490N	.45	SN74174N	1.00
SN7412N	.25	SN7491N	.59	SN74175N	1.00
SN7413N	.40	SN7492N	.43	SN74176N	.79
SN7414N	.70	SN7493N	.43	SN74177N	.79
SN7416N	.25	SN7494N	.65	SN74179N	1.95
SN7417N	.25	SN7495N	.65	SN74180N	.79
SN7420N	.20	SN7496N	.65	SN74181N	1.95
SN7421N	.29	SN7497N	3.00	SN74182N	.79
SN7422N	.39	SN74100N	1.25	SN74184N	1.95
SN7423N	.25	SN74107N	.35	SN74185N	1.95
SN7425N	.29	SN74109N	.59	SN74186N	9.95
SN7426N	.29	SN74116N	1.95	SN74188N	3.95
SN7427N	.25	SN74121N	.35	SN74190N	1.25
SN7429N	.39	SN74122N	.39	SN74191N	1.25
SN7430N	.20	SN74123N	.59	SN74192N	.79
SN7432N	.25	SN74125N	.49	SN74193N	.79
SN7437N	.25	SN74126N	.49	SN74194N	.89
SN7438N	.40	SN74132N	.75	SN74195N	.69
SN7439N	.25	SN74136N	.75	SN74196N	.89
SN7440N	.20	SN74141N	.79	SN74197N	.89
SN7441N	.89	SN74142N	2.95	SN74198N	1.49
SN7442N	.59	SN74143N	2.95	SN74199N	1.49
SN7443N	.75	SN74144N	2.95	SN74S200	4.95
SN7444N	.75	SN74145N	.79	SN74251N	.99
SN7445N	.75	SN74147N	1.95	SN74279N	.79
SN7446N	.69	SN74148N	1.29	SN74283N	2.25
SN7447N	.59	SN74150N	1.25	SN74284N	3.95
SN7448N	.79	SN74151N	.59	SN74285N	3.95
SN7450N	.20	SN74152N	.59	SN74365N	.69
SN7451N	.20	SN74153N	.59	SN74366N	.69
SN7453N	.20	SN74154N	1.50	SN74367N	.69
SN7454N	.20	SN74155N	.79	SN74368N	.69
SN7459A	.25	SN74156N	.79	SN74390N	1.95
SN7460N	.20	SN74157N	.65	SN74393N	1.95

CMOS

Part	Price	Part	Price	Part	Price
CD4000	.39			CD4070	.55
CD4001	.39			CD4071	.49
CD4002	.39	CD4028	.89	CD4072	.49
CD4006	1.19	CD4029	1.49	CD4076	1.39
CD4007	.25	CD4030	.49	CD4081	.39
CD4009	.49	CD4035	.99	CD4082	.39
CD4010	.49	CD4040	1.49	CD4093	.99
CD4011	.39	CD4041	1.49	CD4098	1.19
CD4012	.25	CD4042	.99	MC14409	14.95
CD4013	.49	CD4043	.89	MC14410	14.95
CD4014	1.39	CD4044	.89	MC14411	14.95
CD4015	1.19	CD4046	1.79	MC14419	4.95
CD4016	.59	CD4047	2.50	MC14433	13.95
CD4017	1.19	CD4048	1.35	MC14506	.75
CD4018	.99	CD4049	.49	MC14507	.99
CD4019	.49	CD4050	.69	MC14562	11.95
CD4020	1.19	CD4051	1.19	MC14583	2.49
CD4021	1.39	CD4053	1.19	CD4508	3.95
CD4022	1.19	CD4056	2.95	CD4510	1.39
CD4023	.29	CD4059	9.95	CD4511	1.29
CD4024	.79	CD4060	1.49	CD4515	2.95
CD4025	.23	CD4066	.79	CD4518	1.29
CD4026	2.95	CD4068	.39	CD4520	1.29
CD4027	.69	CD4069	.45	CD4566	2.25

74C00

Part	Price	Part	Price	Part	Price
74C00	.39			74C163	2.49
74C02	.39			74C164	2.49
74C04	.45	74C85	2.49	74C173	2.60
74C08	.49	74C90	1.95	74C192	2.49
74C10	.39	74C93	1.95	74C193	2.49
74C14	1.95	74C95	1.95	74C195	2.49
74C20	.39	74C107	1.25	74C922	7.95
74C30	.39	74C151	2.90	74C923	6.25
74C42	1.95	74C154	3.00	74C925	8.95
74C48	2.49	74C157	2.15	74C926	8.95
74C73	.89	74C160	2.49	80C95	1.50
74C74	.89	74C161	2.49	80C97	1.50

LINEAR

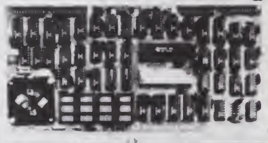
Part	Price	Part	Price	Part	Price
78MG	1.75			LM710N	.79
LM106H	.99			LM711N	.39
LM300H	.80			LM723N/H	.55
LM301CN/H	.35	LM340K-18	1.35	LM733N	1.00
LM302H	.75	LM340K-24	1.35	LM739N	1.19
LM304H	1.00	LM340T-5	1.25	LM741CN/H	.35
LM305H	.60	LM340T-6	1.25	LM741-14N	.39
LM307CN/H	.35	LM340T-8	1.25	LM747N/H	.79
LM308CN/H	1.00	LM340T-12	1.25	LM748N/H	.39
LM309H	1.10	LM340T-15	1.25	LM1310N	1.95
LM309K	1.25	LM340T-18	1.25	LM1458CN/H	.59
LM310CN	1.95	LM340T-24	1.25	MC1488N	1.95
LM311N/H	.90	LM358N	1.00	MC1489N	1.95
LM312H	1.95	LM370N	1.95	LM1496N	.95
LM317K	6.50	LM373N	3.25	LM1556V	1.75
LM318CN/H	1.50	LM377N	4.00	MC1741SCP	3.00
LM319N	1.30	LM380N	1.25	LM2111N	1.95
LM320K-5	1.35	LM380CN	.99	LM2901N	2.95
LM320K-5.2	1.35	LM381N	1.79	LM3053N	1.50
LM320K-12	1.25	LM382N	1.79	LM3065N	1.49
LM320K-15	1.35	NE501N	8.00	LM3900N(3401)	.59
LM320K-18	1.35	NE510A	6.00	LM3905N	1.49
LM320K-24	1.35	NE529A	4.95	LM3909N	1.25
LM320T-5	1.25	NE531H/V	3.95	MC5558V	.59
LM320T-5.2	1.25	NE536T	6.00	8038B	4.95
LM320T-8	1.25	NE540	6.00	LM75450N	.49
LM320T-12	1.25	NE544N	4.95	75451CN	.39
LM320T-15	1.25	NE550N	1.30	75452CN	.39
LM320T-18	1.25	NE555V	.39	75453CN	.39
LM320T-24	1.25	NE556N	.99	75454CN	.39
LM323K-5	5.95	NE560B	5.00	75491CN	.79
LM324N	1.49	NE562B	5.00	75492CN	.89
LM339N	.99	NE565N/H	1.25	75493N	.89
LM340K-5	1.35	NE566CN	1.75	75494CN	.89
LM340K-6	1.35	NE567V/H	.99	RC4136	1.25
LM340K-8	1.35	NE570N	4.95	RC4151	3.95
LM340K-12	1.35	LM703CN/H	.69	RC4194	4.95
LM340K-15	1.35	LM709N/H	.29	RC4195	4.49

74LS00TTL

Part	Price	Part	Price	Part	Price
74LS00	.35			74LS139	1.05
74LS01	.35	74LS51	.29	74LS151	1.05
74LS02	.35	74LS54	.29	74LS155	1.05
74LS03	.35	74LS55	.29	74LS157	1.05
74LS04	.42	74LS73	.54	74LS160	1.15
74LS05	.42	74LS74	.54	74LS161	1.39
74LS08	.35	74LS75	.71	74LS162	1.25
74LS09	.42	74LS76	.54	74LS163	1.39
74LS10	.35	74LS78	.49	74LS164	1.50
74LS11	.75	74LS83	1.05	74LS175	1.25
74LS13	.59	74LS85	1.50	74LS181	2.49
74LS14	1.25	74LS86	.54	74LS190	1.39
74LS15	.35	74LS90	.71	74LS191	1.39
74LS20	.35	74LS92	.90	74LS192	1.39
74LS21	.35	74LS93	.90	74LS193	1.39
74LS22	.35	74LS95	.99	74LS194	1.39
74LS26	.35	74LS96	1.15	74LS195	1.39
74LS27	.35	74LS107	.54	74LS253	1.25
74LS28	.35	74LS109	.54	74LS257	1.05
74LS30	.35	74LS112	.54	74LS258	1.75
74LS32	.42	74LS123	1.50	74LS260	.83
74LS37	.45	74LS125	1.05	74LS279	.90
74LS40	.35	74LS132	.99	74LS367	.75
74LS42	1.05	74LS136	.59	74LS368	.75
74LS47	1.05	74LS138	1.05	74LS670	2.49

Cromemco incorporated

Z80-4MHz Single Card Computer



Cromemco's Single Card Computer is a complete computer which brings the power of the Z-80 and the flexibility of the S-100 bus to the dedicated computer environment.

The card offers 4 MHz operation, 8K bytes of on-board 2716 PROM, and 1K byte of static RAM memory. This stand-alone card also provides an RS-232 (or 20mA current loop) serial interface with programmable baud rates to 76,800, vectored interrupts, 24 bits of bidirectional parallel I/O, and 5 programmable timers. Only a power supply and PROM software are required for operation. The Single Card Computer is assembled and tested (Model SCC-W) for $450. The Monitor and Control BASIC are available in two ROM's (Model MCB-216) for $98.

SCC-W (Assembled) **$450.00**

Processor: 4MHz Z-80
Instruction Set: 158 instructions incl. the 78 instructions of the 8080
ROM Capacity: 8K Bytes located from address 0000 to 1FFF
ROM Type: Intel 2716 PROM or equivalent
RAM Capacity: 1K Bytes located from address 2000 to 23FF
RAM Type: 4045 Static

Serial I/O Ports:
I/O Levels: RS 232 or 20mA current loop
Baud Rates: 110 to 76,800 (software selectable)
Parallel Ports:
Input Port: 24 bits bidirectional
Output Port: 24 bits bidirectional
Input Load: One TTL equivalent
Output Drive: 20 TTL loads
Interval Timers:
Number of timers: 5

Timer range: 0.16-32 milliseconds (software set)
Timer resolution: 64 microseconds
Vectored Interrupts:
No. of restart locations (Z-80 mode): 65,536
General Information:
UART type: 5601 Bus: S-100
Power requirements: +8V at 1.4A; +18V at 70mA; -18V at 25mA
Operating environment: 0-55°C

DISCRETE LEDS

XC556R	.200"	red	5/$1	MV50	.085"	red	6/$1	XC111R	.190"	red	5/$1
XC556G	.200"	green	4/$1	XC209R	.125"	red	5/$1	XC111G	.190"	green	4/$1
XC556Y	.200"	yellow	4/$1	XC209G	.125"	green	4/$1	XC111Y	.190"	yellow	4/$1
XC556C	.200"	clear	4/$1	XC209Y	.125"	yellow	4/$1	XC111C	.190"	clear	4/$1
XC22R	.200"	red	5/$1	XC526R	.185"	red	5/$1				
XC22G	.200"	green	4/$1	XC526G	.185"	green	4/$1	INFRA-RED LED			
XC22Y	.200"	yellow	4/$1	XC526Y	.185"	yellow	4/$1	¼"x¼"x1/16"		flat	5/$1
MV10B	.170"	red	4/$1	XC526C	.185"	clear	4/$1				

DISPLAY LEDS

TYPE	POLARITY	HT	PRICE	TYPE	POLARITY	HT	PRICE
MAN 1	Common Anode-red	.270	2.95	MAN 6730	Common Anode-red ± 1	560	.99
MAN 2	5 x 7 Dot Matrix-red	.300	4.95	MAN 6740	Common Cathode-red-D.D	560	.99
MAN 3	Common Cathode-red	.125	.25	MAN 6750	Common Cathode-red ± 1	560	.99
MAN 4	Common Cathode-red	.187	1.95	MAN 6760	Common Anode-red	560	.99
MAN 7G	Common Anode-green	.300	1.25	MAN 6780	Common Cathode-red	560	.99
MAN 7Y	Common Anode-yellow	.300	.99	DL701	Common Anode-red ± 1	300	.99
MAN 72	Common Anode-red	.300	.75	DL704	Common Cathode-red	300	.99
MAN 74	Common Cathode-red	.300	1.25	DL707	Common Anode-red	300	.99
MAN 82	Common Anode-yellow	.300	.49	DL728	Common Cathode-red	500	1.49
MAN 84	Common Cathode-yellow	.300	.99	DL741	Common Anode-red	600	1.25
MAN 3620	Common Anode-orange	.300	.99	DL746	Common Anode-red ± 1	630	1.49
MAN 3630	Common Anode-orange ± 1	.300	.99	DL747	Common Anode-red	600	1.49
MAN 3640	Common Cathode-orange	.300	.99	DL749	Common Cathode-red ± 1	630	1.49
MAN 4610	Common Anode-orange	.300	.99	DL750	Common Cathode-red	600	1.49
MAN 4640	Common Cathode-orange	.400	.99	DL338	Common Cathode-red	.110	.35
MAN 4710	Common Anode-red	.400	.99	FND70	Common Cathode	250	.69
MAN 4730	Common Anode-red ± 1	.400	.99	FND358	Common Cathode ± 1	357	.99
MAN 4740	Common Cathode-red	.400	.99	FND359	Common Cathode	357	.75
MAN 4810	Common Anode-yellow	.400	.99	FND503	Common Cathode(FND500)	500	.99
MAN 4840	Common Cathode-yellow	.400	.99	FND507	Common Anode (FND510)	500	.99
MAN 6610	Common Anode-orange-D.D.	.560	.99	5082-7730	Common Cathode-red	300	.99
MAN 6630	Common Anode-orange ± 1	.560	.99	HDSP-3400	Common Anode-red	800	1.50
MAN 6640	Common Cathode-orange-D.D.	.560	.99	HDSP-3403	Common Cathode red	800	1.50
MAN 6650	Common Cathode-orange ± 1	.560	.99	5082-7300	4 x 7 Sgl. Digit-RHDP	600	19.95
MAN 6660	Common Anode-orange	.560	.99	5082-7302	4 x 7 Sgl. Digit-LHDP	600	19.95
MAN 6680	Common Cathode-orange	.560	.99	5082-7304	Overrange character (± 1)	600	15.00
MAN 6710	Common Anode-red-D.D.	.560	.99	5082-7340	4 x 7 Sgl. Digit-Hexadecimal	600	22.50

RCA LINEAR

CA3013T	2.15	CA3082N	2.00
CA3023T	3.25	CA3083N	1.60
CA3035T	2.48	CA3086N	.85
CA3039T	1.35	CA3089N	3.75
CA3046N	1.30	CA3130T	1.39
CA3059N	3.25	CA3140T	1.25
CA3060N	3.25	CA3160T	1.25
CA3080T	1.25	CA3401N	.59
CA3081N	2.00	CA3600N	3.50

CALCULATOR CHIPS/DRIVERS

MM5725	$2.95
MM5738	2.95
DM8864	2.00
DM8865	1.00
DM8887	.75
DM8889	.75
9374 7-seg. LED driver	1.50
MM5309	4.95

CLOCK CHIPS

MM5309	4.95
MM5311	4.95
MM5312	4.95
MM5314	4.95
MM5316	6.95
MM5318	9.95
MM5369	2.95
MM5387/1998a	4.95
CT 7001	6.95

MOTOROLA

MC1408L7	4.95
MC1408L8	5.75
MC1439L	2.95
MC3022P	2.95
MC3061P	3.50
MC4016(74416)	7.50
MC4024P	3.95
MC4040P	6.95
MC4044P	4.50

LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100
8 pin LP	.17	.16	.15
14 pin LP	.20	.19	.18
16 pin LP	.22	.21	.20
18 pin LP	.29	.28	.27
20 pin LP	.34	.32	.30
22 pin LP	.37	.36	.35
24 pin LP	.38	.37	.36
28 pin LP	.45	.44	.43
36 pin LP	.60	.59	.58
40 pin LP	.63	.62	.61

SOLDERTAIL STANDARD (TIN)

	1-24	25-49	50-100
14 pin ST	.27	.25	.24
16 pin ST	.30	.27	.25
18 pin ST	.35	.32	.30
24 pin ST	.49	.45	.42
28 pin ST	.99	.90	.81
36 pin ST	1.39	1.26	1.15
40 pin ST	1.59	1.45	1.30

SOLDERTAIL (GOLD) STANDARD

	1-24	25-49	50-100
8 pin SG	.39	.35	.31
14 pin SG	.49	.45	.41
16 pin SG	.54	.49	.44
18 pin SG	.59	.53	.48
24 pin SG	.79	.75	.69
28 pin SG	1.10	1.00	.90
36 pin SG	1.65	1.40	1.26
40 pin SG	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

	1-24	25-49	50-100
8 pin WW	.59	.54	.49
10 pin WW	.69	.63	.58
14 pin WW	.79	.73	.67
16 pin WW	.85	.77	.70
18 pin WW	.99	.90	.81
20 pin WW	1.19	1.08	.99
22 pin WW	1.49	1.35	1.23
24 pin WW	1.39	1.26	1.14
28 pin WW	1.69	1.53	1.38
36 pin WW	2.19	1.99	1.79
40 pin WW	2.29	2.09	1.89

1/4 WATT RESISTOR ASSORTMENTS – 5%

ASST. 1	5 ea	10 OHM / 27 OHM	12 OHM / 33 OHM	15 OHM / 39 OHM	18 OHM / 47 OHM	22 OHM / 56 OHM	50 PCS	$1.75
ASST. 2	5 ea	68 OHM / 180 OHM	82 OHM / 220 OHM	100 OHM / 270 OHM	120 OHM / 330 OHM	150 OHM / 390 OHM	50 PCS	1.75
ASST. 3	5 ea	470 OHM / 1.2K	560 OHM / 1.5K	680 OHM / 1.8K	820 OHM / 2.2K	1K / 2.7K	50 PCS	1.75
ASST. 4	5 ea	3.3K / 8.2K	3.9K / 10K	4.7K / 12K	5.6K / 15K	6.8K / 18K	50 PCS	1.75
ASST. 5	5 ea	22K / 56K	27K / 68K	33K / 82K	39K / 100K	47K / 120K	50 PCS	1.75
ASST. 6	5 ea	150K / 390K	180K / 470K	220K / 560K	270K / 680K	330K / 820K	50 PCS	1.75
ASST. 7	5 ea	1M / 2.7M	1.2M / 3.3M	1.5M / 3.9M	1.8M / 4.7M	2.2M / 5.6M	50 PCS	1.75

ASST. 8R Includes Resistor Assortments 1-7 (350 PCS.) **$9.95 ea.**

TELEPHONE/KEYBOARD CHIPS

AY-5-9100	Push Button Telephone Dialler	$14.95
AY-5-9200	Repertory Dialler	14.95
AY-5-9500	CMOS Clock Generator	4.95
AY-5-2376	Keyboard Encoder (88 keys)	14.95
HD0165	Keyboard Encoder (16 keys)	7.95
74C922	Keyboard Encoder (16 keys)	7.95
74C923	Keyboard Encoder (20 keys)	6.25

ICM CHIPS

ICM7045	CMOS Precision Timer	24.95
ICM7205	CMOS LED Stopwatch/Timer	19.95
ICM7207	Oscillator Controller	7.50
ICM7208	Seven Decade Counter	19.95
ICM7209	Clock Generator	6.95

NMOS READ ONLY MEMORIES

MCM6571	128 X 9 X 7 ASCII Shifted with Greek	13.50
MCM6574	128 X 9 X 7 Math Symbol & Pictures	13.50
MCM6575	128 X 9 X 7 Alpha Control Char Gen	13.50

MISCELLANEOUS

TL074CN	Quad Low Noise bi-fet Op Amp	2.49
TL494CN	Switching Regulator	4.49
TL496CP	Single Switching Regulator	1.75
11C90	Divide 10/11 Prescaler	19.95
95H90	Hi-Speed Divide 10/11 Prescaler	11.95
4N33	Photo-Darlington Opto-Isolator	3.95
MK50240	Top Octave Freq. Generator	17.50
DS0026CH	5Mhz 2-phase MOS clock driver	3.75
TIL308	.27" red num. display w/integ. logic chip	10.95
MM5320	TV Camera Sync. Generator	14.95
MM5330	4½ Digit DPM Logic Block (Special)	3.95
LD110/111	3½ Digit A/D Converter Set	25.00/set
MC14433P	3½ Digit A/D Converter	13.95

LITRONIX ISO-LIT 1
Photo Transistor Opto-Isolator
(Same as MCT 2 or 4N25)
49¢ each

SN 76477
SOUND GENERATOR
Generates Complex Sounds
Low Power - Programmable
$3.95 each

TV GAME CHIP AND CRYSTAL
AY-3-8500-1 and 2.01 MHZ Crystal (Chip & Crystal includes score display, 6 games and select angles, etc. **7.95/set**

EXAR

XR205	$8.40			XR2242CP	1.50
XR210	4.40			XR2264	4.25
XR215	4.40			XR2556	3.20
XR320	1.55			XR2567	2.99
XR-L555	1.50	JE2206KB	19.95	XR3403	1.25
XR555	.39	XR1800	3.20	XR4136	1.25
XR556	.99	XR2206	4.40	XR4151	3.95
XR567CP	.99	XR2207	3.85	XR4194	4.95
XR567CT	1.25	XR2208	5.20	XR4202	3.60
XR1310P	1.95	XR2209	1.75	XR4212	2.05
XR1468CN	3.85	XR2211	5.25	XR4558	.75
XR1488	1.95	XR2212	4.35	XR4739	1.15
XR1489	1.95	XR2240	3.45	XR4741	1.47

DIODES

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
				1N4002	100 PIV	1 AMP	12/1.00
				1N4003	200 PIV	1 AMP	12/1.00
1N746	3.3	400m	4/1.00	1N4004	400 PIV	1 AMP	12/1.00
1N751	5.1	400m	4/1.00	1N4005	600 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4006	800 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N4007	1000 PIV	1 AMP	10/1.00
1N754	6.8	400m	4/1.00	1N3600	50	200m	6/1.00
1N757	9.0	400m	4/1.00	1N4148	75	10m	15/1.00
1N759	12.0	400m	4/1.00	1N4154	35	10m	12/1.00
1N959	8.2	400m	4/1.00	1N4733	5.1	1w	28
1N965	15	400m	4/1.00	1N4734	5.6	1w	28
1N5232	5.6	500m	28	1N4735	6.2	1w	28
1N5234	6.2	500m	28	1N4736	6.8	1w	28
1N5235	6.8	500m	28	1N4738	8.2	1w	28
1N5236	7.5	500m	28	1N4742	12	1w	28
1N5242	12	500m	28	1N4744	15	1w	28
1N5245	15	500m	28	1N1183	50 PIV	35 AMP	1.60
1N456	25	40m	6/1.00	1N1184	100 PIV	35 AMP	1.70
1N458	150	7m	6/1.00	1N1185	150 PIV	35 AMP	1.70
1N485A	180	10m	5/1.00	1N1186	200 PIV	35 AMP	1.80
1N4001	50 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR(2N1849)	$1.95
C36M	35A @ 600V	SCR	1.95
2N2328	1.6A @ 300V	SCR	.50
MDA 980-1	12A @ 50V	FW BRIDGE REC	1.95
MDA 980-3	12A @ 200V	FW BRIDGE REC	1.95

TRANSISTORS

C106B1	.50			2N3904	4/1.00
MPSA05	.30	2N3055	.89	2N3905	4/1.00
MPSA06	5/1.00	MJE3055	1.00	2N3906	4/1.00
TIS97	6/1.00	2N3392	5/1.00	2N4013	3/1.00
TIS98	6/1.00	2N3398	5/1.00	2N4123	6/1.00
40409	1.75	PN3567	3/1.00	PN4249	4/1.00
40410	1.75	PN3568	4/1.00	PN4250	4/1.00
40673	1.75	PN3569	4/1.00	2N4400	4/1.00
2N918	4/1.00	MPS3638A	5/1.00	2N4401	4/1.00
2N2219A	2/1.00	MPS3702	5/1.00	2N4402	4/1.00
2N2221A	4/1.00	2N3704	5/1.00	2N4403	4/1.00
2N2222A	5/1.00	MPS3704	5/1.00	2N4409	5/1.00
PN2222 Plastic	7/1.00	2N3705	5/1.00	2N5086	4/1.00
2N2369A	4/1.00	MPS3705	5/1.00	2N5087	4/1.00
MPS2369	5/1.00	2N3706	5/1.00	2N5088	4/1.00
2N2484	4/1.00	MPS3706	5/1.00	2N5089	4/1.00
2N2906	4/1.00	2N3707	5/1.00	2N5129	5/1.00
2N2907	5/1.00	2N3711	5/1.00	PN5134	5/1.00
		2N3724A	.65	PN5138	5/1.00
2N2925	5/1.00	2N3725A	1.00	2N5139	5/1.00
MJE2955	1.25	2N3772	2.25	2N5210	5/1.00
2N3053	2/1.00	2N3823	1.00	2N5449	3/1.00
		2N3903	5/1.00	2N5951	3/1.00

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-99	100-		1-9	10-99	100-
10 pf	05	04	03	001µF	05	04	.035
22 pf	05	04	03	0047µF	05	04	035
47 pf	05	.04	03	01µF	05	04	035
100 pf	05	.04	03	022µF	06	05	04
220 pf	05	04	03	047µF	06	05	04
470 pf	05	04	.035	1µF	12	.09	075
100 VOLT MYLAR FILM CAPACITORS							
001mf	12	10	.07	.022mf	13	11	08
.0022	12	10	.07	.047mf	21	17	13
0047mf	.12	10	07	1mf	27	.23	17
01mf	12	10	07	.22mf	33	.27	22
+20% DIPPED TANTALUMS (SOLID) CAPACITORS							
1/35V	.39	.31	.25	1.5/35V	.41	.33	.26
15/35V	.39	.31	.25	2.2/35V	.51	.41	.33
22/35V	.39	.31	.25	3.3/25V	.53	.43	.34
33/35V	.39	.31	.25	4.7/25V	.63	.51	.41
47/35V	.39	.31	.25	6.8/25V	.79	.63	.50
68/35V	.39	.31	.25	15/25V	1.39	1.12	.89
1.0/35V	.39	.31	.25	22/6V	.79	.63	.50
MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS							
Axial Lead				Radial Lead			
47/50V	.15	.13	10	47/25V	15	13	10
1.0/50V	.16	.14	11	47/50V	16	.14	11
3.3/50V	14	.12	09	1.0/16V	15	.13	10
4.7/25V	15	.13	10	1.0/25V	16	.14	.11
10/25V	15	.13	10	1.0/50V	16	.14	11
10/50V	16	.14	12	4.7/16V	15	.13	.10
22/25V	17	.15	12	4.7/25V	15	13	10
22/50V	24	.20	18	4.7/50V	.16	14	.11
47/25V	.19	17	.15	10/16V	.14	12	09
47/50V	.25	.21	.19	10/25V	15	13	.10
100/25V	24	.20	18	10/50V	.16	14	12
100/50V	.35	.30	28	47/50V	24	21	.19
220/25V	.32	28	.25	100/16V	19	.15	.14
220/50V	.45	.41	38	100/25V	24	.20	18
470/25V	.33	.29	.27	100/50V	.35	.30	28
1000/16V	55	50	.45	220/16V	23	17	16
2200/16V	70	62	.55	470/25V	.31	.28	26

$10.00 Min. Order — U.S. Funds Only
Calif. Residents Add 6% Sales Tax
Postage — Add 5% plus $1 Insurance (if desired)

Spec Sheets — 25¢
1980 Catalog Available — Send 41¢ stamp

FREE 1980 CATALOG

Jameco ELECTRONICS

PHONE ORDERS WELCOME (415) 592-8097

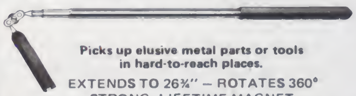
MAIL ORDER ELECTRONICS – WORLDWIDE
1355 SHOREWAY ROAD, BELMONT, CA 94002
PRICES SUBJECT TO CHANGE

4-80

MINIATURE SCREWDRIVER

RUGGED CONSTRUCTION
1000's OF USES – 1/8" TIP .... 2/$1.00

MAGNETIC RETRIEVER TOOL



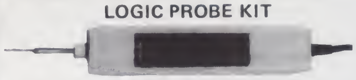
Picks up elusive metal parts or tools in hard-to-reach places.

EXTENDS TO 26¾" – ROTATES 360°
STRONG, LIFETIME MAGNET

MRT-2281 .... $3.95

CSC CONTINENTAL SPECIALTIES

LOGIC PROBE KIT



SPECIFICATIONS

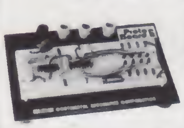
Input Impedance: 300,000 Ohms.
Thresholds: "Lo" 30%Vcc – "Hi" 70%Vcc
Maximum Speed: 300 nsec., 1.5MHz
Input Protection: ±50VDC continuous 117VAC for 15 sec.
Power: 30mA @5V – 40mA @ 15V – 25V max. reverse voltage protected; 36" cable with color coded clips included.
Operating Temp.: 0-50°C.
Dimensions: 5.8L x 1.0W x 0.7D in. (147 x 25 x 18mm)
Weight: 30 oz. (85 gm)

LPK-1 .... $21.95/Kit

Proto Clips

14-PIN CLIP	PC-14	$ 4.50
16-PIN CLIP	PC-16	$ 4.75
24-PIN CLIP	PC-24	$10.00
40-PIN CLIP	PC-40	$16.00

Proto Boards

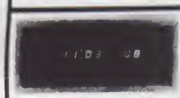


PB-6	$17.95
PB-100	19.95
PB-101	22.95
PB-102	26.95
PB-103	44.95
PB-104	55.95
PB-203	99.95
PB-203A	155.00
PB-203A-Kit	131.00

Jumbo 6-Digit Clock Kit

* Four .630"ht. and two .300"ht. common anode displays
* Uses MM5314 clock chip
* Switches for hours, minutes and hold functions
* Hours easily viewable to 30 feet
* Simulated walnut case
* 115 VAC operation
* 12 or 24 hour operation
* Includes all components, case and wall transformer
* Size: 6¾ x 3⅛ x 1¾

JE747 .... $29.95



- Bright .300 ht. comm. cathode display
- Uses MM5314 clock chip
- Switches for hours, minutes and hold modes
- Hrs. easily viewable to 20 ft.
- Simulated walnut case
- 115 VAC operation
- 12 or 24 hr. operation
- Incl. all components, case & wall transformer
- Size: 6¾" x 3-1/8" x 1¾"

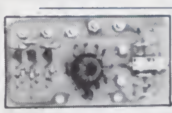
JE701

6-Digit Clock Kit $19.95

Regulated Power Supply

Uses LM309K. Heat sink provided. PC board construction. Provides a solid 1 amp @ 5 volts. Can supply up to ±5V, ±9V and ±12V with JE205 Adapter. Includes components, hardware and instructions. Size: 3½" x 5" x 2"H

JE200 .... $14.95



ADAPTER BOARD
–Adapts to JE200–
±5V, ±9V and ±12V

DC/DC converter with +5V input. Toriodal hi-speed switching XMFR. Short circuit protection. PC board construction. Piggy-back to JE 200 board. Size: 3½" x 2" x 9/16"H

JE205 .... $12.95

MICROPROCESSOR COMPONENTS

8080A/8080A SUPPORT DEVICES

8080A	CPU	$ 7 95
8212	8-Bit Input/Output	3.25
8214	Priority Interrupt Control	5 95
8216	Bi-Directional Bus Driver	3 49
8224	Clock Generator/Driver	3.95
8226	Bus Driver	3 49
8228	System Controller/Bus Driver	4 95
8238	System Controller	5 95
8251	Prog Comm 1/0 (USART)	7 95
8253	Prog Interval Timer	14 95
8255	Prog Periph 1/0 (PPI)	9 95
8257	Prog DMA Control	19 95
8259	Prog Interrupt Control	19 95

6800/6800 SUPPORT DEVICES

MC6800	MPU	$14 95
MC6802CP	MPU with Clock and Ram	24 95
MC6810API	128X8 Static Ram	5 95
MC6821	Periph. Inter. Adapt (MC6820)	7 49
MC6828	Priority Interrupt Controller	12 95
MC6830L8	1024X8 Bit ROM (MC68A30-8)	14.95
MC6850	Asynchronous Comm. Adapter	7 95
MC6852	Synchronous Serial Data Adapt	9 95
MC6860	0-600 bps Digital MODEM	12 95
MC6862	2400 bps Modulator	14.95
MC6880A	Quad 3-State Bus Trans (MC8T26)	2.25

MICROPROCESSOR CHIPS—MISCELLANEOUS

Z80(780C)	CPU	$13.95
Z80A(780-1)	CPU	15.95
CDP1802	CPU	19 95
2650	MPU	19 95
6502	CPU	11 95
8035	8-Bit MPU w/clock, RAM, 1/0 lines	19 95
P8085	CPU	19 95
TMS9900JL	16-Bit MPU w/hardware, multiply & divide	49.95

SHIFT REGISTERS

MM500H	Dual 25 Bit Dynamic	$ 50
MM503H	Dual 50 Bit Dynamic	.50
MM504H	Dual 16 Bit Static	.50
MM506H	Dual 100 Bit Static	50
MM510H	Dual 64 Bit Accumulator	50
MM5016H	500/512 Bit Dynamic	89
2504T	1024 Dynamic	3 95
2518	Hex 32 Bit Static	4 95
2522	Dual 132 Bit Static	2 95
2524	512 Static	99
2525	1024 Dynamic	2 95
2527	Dual 256 Bit Static	2.95
2528	Dual 250 Static	4 00
2529	Dual 240 Bit Static	4.00
2532	Quad 80 Bit Static	2 95
3341	Filo	6 95
74LS670	4X4 Register File (TriState)	2 49

UART'S

A-Y-5-1013	30K BAUD	5 95

MICROPROCESSOR MANUALS

M-Z80	User Manual	$7 50
M-CDP1802	User Manual	7.50
M-2650	User Manual	5 00

ROM'S

2513(2140)	Character Generator(upper case)	$9 95
2513(3021)	Character Generator(lower case)	9 95
2516	Character Generator	10 95
MM5230N	2048-Bit Read Only Memory	1 95

RAM'S

1101	256X1	Static	$1 49
1103	1024X1	Dynamic	99
2101(8101)	256X4	Static	3 95
2102	1024X1	Static	1 75
21L02	1024X1	Static	1 95
2111(8111)	256X4	Static	3 95
2112	256X4	Static MOS	4 95
2114	1024X4	Static 450ns	7 95
2114L	1024X4	Static 450ns low power	10 95
2114-3	1024X4	Static 300ns	10 95
2114L-3	1024X4	Static 300ns low power	11 95
5101	256X4	Static	7 95
5280/2107	4096X1	Dynamic	4 95
7489	16X4	Static	1.75
74S200	256X1	Static Tristate	4 95
93421	256X1	Static	2 95
UPD414 (MK4027)	4K	Dynamic 16 pin	4 95
UPD416 (MK4116)	16K	Dynamic 16 pin 250ns	9 95
TMS4044-45NL	4K	Static	14 95
TMS4045	1024X4	Static	14 95
2117	16,384X1	Dynamic 350ns (house marked)	9 95
MM5262	2KX1	Dynamic	4/1.00

PROM'S

1702A	2048	FAMOS	$5 95
2716INTEL	16K*	EPROM	59 95
TMS2516 (2716)	16K*	EPROM	39.95
	*Requires single +5V power supply		
TMS2532	4KX8	EPROM	89 95
2708	8K	EPROM	10 95
2716 T I	16K**	EPROM	29 95
	**Requires 3 voltages, —5V, +5V, +12V		
5203	2048	FAMOS	14 95
6301-1(7611)	1024	Tristate Bipolar	3.49
6330-1(7602)	256	Open C Bipolar	2.95
82S23	32X8	Open Collector	3 95
82S115	4096	Bipolar	19 95
82S123	32X8	Tristate	3 95
74186	512	TTL Open Collector	9 95
74188	256	TTL Open Collector	3.95
74S287	1024	Static	2 95

JE600 HEXADECIMAL ENCODER KIT

FEATURES:
- Full 8 bit latched output for microprocessor use
- 3 User Define keys with one being bi-stable operation
- Debounce circuit provided for all 19 keys
- LED readout to verify entries
- Easy interfacing with standard 16 pin 1C connector
- Only +5VDC required for operations

FULL 8 BIT LATCHED OUTPUT—19 KEYBOARD

The JE600 Encoder Keyboard provides two separate hexadecimal digits produced from sequential key entries to allow direct programming for 8 bit microprocessor or 8 bit memory circuits Three (3) additional keys are provided for user operations with one having a bistable output available The outputs are latched and monitored with LED readouts Also included is a key entry strobe

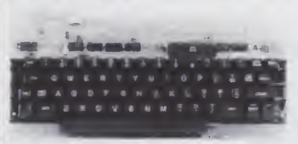
JE600 .... $59.95
Hexadecimal Keypad only $14.95

DIGITAL THERMOMETER KIT

- Dual sensors—switching control for indoor/outdoor or dual monitoring
- Continuous LED .8" ht. display
- Range: -40°F to 199°F / -40°C to 100°C
- Accuracy: ±1° nominal
- Set for Fahrenheit or Celsius reading
- Sim. walnut case - AC wall adapter incl.
- Size: 3-1/4"H x 6-5/8"W x 1-3/8"D

JE300 .... $39.95

62-Key ASCII Encoded Keyboard Kit



The JE610 62-Key ASCII Encoded Keyboard Kit can be interfaced into most any computer system. The JE610 Kit comes complete with an industrial grade keyboard switch assembly (62 keys), IC's, sockets, connector, electronic components and a double-sided printed wiring board. The keyboard assembly requires +5V @ 150mA and –12V @ 10mA for operation.

FEATURES:
- 60 Keys generate the full 128 characters, upper and lower case ASCII set
- Fully buffered
- 2 user-define keys provided for custom applications
- Caps lock for upper case only alpha characters
- Utilizes a 2376 (40 pin) encoder read only memory chip
- Outputs directly compatible with TTL/DTL or MOS logic arrays
- Easy interfacing with a 16-pin dip or 18-pin edge connector

JE610 ..... $79.95
62-Key Keyboard only . . $34.95

HICKOK LX303 Portable LCD Digital Multimeter

- Big ½ inch high, 3½ Digit Liquid Crystal Display
- Over 200 Hours of operation with a 9V battery
- Automatic zero, polarity and overrange indication
- 100mV DC F S sensitivity
- 1% basic accuracy
- 19 ranges and functions
- Rugged Cycolac case and removable cover, stores test lead set
- Full one year warranty

SPECIFICATIONS DC volts (5 ranges) 0 1mV to 1000V Accuracy ±0 5% rdg ±0 5% f s Input imped 10MΩ Max input 1kV except 500V on 200mV range
AC Volts (40Hz to 5kHz) 0 1V to 600V Accuracy ±1 0% rdg ±0 5% f s (−2dB max at 5kHz) Max input 600V
Resistance (6 Low Power Ranges) 0 1 to 20MΩ Accuracy ±0 5% rdg ±0 5% f s (±1 5% rdg on 20MΩ range) input protected to 120VAC all ranges
DC Current (6 ranges) 01nA to 100mA Accuracy ±1 0% rdg ±0 5% f s
Dimensions and Weight 5-7/8 x 3-3/8 x 1 3/4 8 oz POWER 9V battery (not included) or Hickok AC adapter
READ RATE 3/sec

LX303 Digital Multimeter $74.95

PART NO	DESCRIPTION	PRICE
RC-3	115V AC ADAPTER	7.50
CC-3	PADDED CARRYING CASE	7.50
VP-10	X10 DC PROBE ADAPTER (Up to 10K)	14 95
VP-40	40kV DC PROBE	35.00
CS-1	10 Amp DC Current Shunt	14.95

$10.00 Min. Order – U.S. Funds Only
Calif. Residents Add 6% Sales Tax
Postage – Add 5% plus $1 Insurance (if desired)

Spec Sheets – 25¢
1980 Catalog Available – Send 41¢ stamp

FREE 1980 CATALOG

Jameco ELECTRONICS ✓41

PHONE ORDERS WELCOME (415) 592-8097

MAIL ORDER ELECTRONICS – WORLDWIDE
1355 SHOREWAY ROAD, BELMONT, CA 94002
PRICES SUBJECT TO CHANGE

4-80

The Incredible "Pennywhistle 103"

$139.95 Kit Only

The Pennywhistle 103 is capable of recording data to and from audio tape without critical speed requirements for the recorder and it is able to communicate directly with another modem and terminal for telephone "hamming" and communications In addition, it is free of critical adjustments and is built with non-precision, readily available parts.

Data Transmission Method .....Frequency-Shift Keying, full-duplex (half-duplex selectable)
Maximum Data Rate ....300 Baud
Data Format ....Asynchronous Serial (return to mark level required between each character)
Receive Channel Frequencies ...2025 Hz for space, 2225 Hz for mark
Transmit Channel Frequencies ..Switch selectable Low (normal) – 1070 space, 1270 mark High – 025 space, 2225 mark
Receive Sensitivity .... 46 dbm accoustically coupled
Transmit Level ....– 15 dbm nominal Adjustable from –6 dbm to –20 dbm
Receive Frequency Tolerance ...Frequency reference automatically adjusts to allow for operation between 1800 Hz and 2400 Hz
Digital Data Interface ....EIA RS-232C or 20 mA current loop (receiver is optoisolated and non-polar)
Power Requirements ....120 VAC, single phase, 10 Watts
Physical ....All components mount on a single 5" by 9" printed circuit board All components included

Requires a VOM, Audio Oscillator, Frequency Counter and/or Oscilloscope to align

TRS-80 16K Conversion Kit

Expand your 4K TRS-80 System to 16K.
Kit comes complete with:
* 8 each UPD416-1 (16K Dynamic Rams) 250NS
* Documentation for conversion

TRS-16K $75.00

ok JUST WRAP™

- 30 AWG wire
- Daisy chain or point-to-point
- No stripping or slitting required—just wrap
- .025" square posts
- Built-in cut off
- Includes 50 ft. wire

Part No.	Wire Color	Price
JW-1-B	Blue	$14.95
JW-1-W	White	14.95
JW-1-Y	Yellow	14.95
JW-1-R	Red	14.95

JUST WRAP™ Replacement Wire

Part No.	Color		Price
R-JW-B	Blue	50 ft. roll	$2.98
R-JW-W	White	50 ft. roll	2.98
R-JW-Y	Yellow	50 ft. roll	2.98
R-JW-R	Red	50 ft. roll	2.98

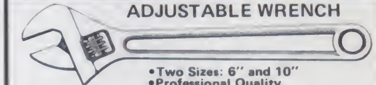
JUST WRAP™ Unwrap Tool $3.49

ok Vacuum Vise

Vacuum-based light-duty vise for small components and assemblies. ABS construction. 1½" jaws, 1¼" travel. Can be permanently installed.

VV-1 ...... $3.49

ADJUSTABLE WRENCH



- Two Sizes: 6" and 10"
- Professional Quality
- Chrome Vanadium Steel

6"– AW-6 . . $4.95 10"– AW-10 . . $6.95

JOYSTICK VIDEO CONTROLLER

FRONT VIEW INSIDE REAR VIEW

- IDEAL FOR ALL VIDEO GAMES OR REMOTE CONTROL PROJECTS
- SMALL CASE SIZE: 1-1/2"H x 2-3/8"W x 4-5/16"L
- 2 MINIATURE POTENTIOMETERS – 40K OHM EACH
- SPST PUSH BUTTON CONTROL
- 5-WIRE CONNECTION CABLE – 5 FEET LONG
- RUGGED PLASTIC CASE

JVC-40 ...... $4.95 each

✓Reader Service – see page 257

Radio Hut Radio Hut Radio Hut Radio Hut

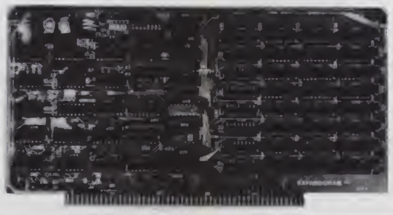
Radio Hut
201 LOCHWOOD MALL • DALLAS, TEXAS 75218
ORDER BY PHONE—214-324-5509

✓46

PLEASE WRITE FOR CATALOG
PRICES SUBJECT TO CHANGE
WITHOUT NOTICE

ORDERING INFORMATION & TERMS: Orders under $15.00 add 75¢ handling. No C.O.D. We accept Visa, MasterCharge, and American Express cards. Tex. Res. add 5% tax. Foreign orders (except Canada) add 20% P&H. 90 Day Money Back Guarantee on all items. Add 5% P&H, maximum $5.00.
ORDER BY PHONE — (214) 324-5509

SD EXPANDORAM



The EXPANDORAM is available in versions from 16K up to 64K, so for a minimum investment you can have a memory system that will grow with your needs. This is a dynamic memory with the invisible on-board refresh, and IT WORKS!

- Bank Selectable
- Phantom
- Power 8VDC, +16VDC, 5 Watts
- Lowest Cost Per Bit
- Uses Major Brand 16K RAMS
- PC Board is doubled solder masked and has silk-screened parts layout
- Extensive documentation clearly written

- Complete kit includes all Sockets for 64K
- Memory access time: 375ns, Cycle time: 500ns.
- No wait states required
- 16K boundaries and Protection, via Dip Switches
- Designed to work with Z-80, 8080, 8085 CPU's

EXPANDORAM 64K Kit (16K Ram)

16K	$239.95
32K	309.95
48K	379.95
64K	449.95

SD'S PROM 100 PROM Programmer Board

The PROM-100 Programmer is a development tool for S-100 Bus computer systems. The Zero Insertion Force Programming Socket extends above the card cage height for easy access to PROM devices. Software verifies PROM erasure, verifies program loading and provides for reading of object file from Disk or PROM and programming into PROM/EPROM. Features include: On-board generated 25vdc Programming pulse, TTL compatible, maximum programming time for 16,389 bits is 100 seconds. Programs: 2708, Intel 2758, 2716, 2732 and TI 2516. DIP Selectable EPROM type.

PROM-100 Board Kit $149.95

SD'S SBC-200 SINGLE BOARD COMPUTER Kit $289.95

S-100 Bus compatible and based on the powerful Z80 microprocessor, the SBC-200 meets the needs of a Z-80 CPU board with many additional features. Ideal for Industrial and control applications. All of the same features that have made the SBC-100 famous, PLUS 4MHz OPERATION. • S-100 Bus Compatible • Z80 Central Processing Unit • 1024 Bytes of Random Access Memory • 8K Bytes of PROM using 2716 • Serial Input/Output Port (with Asynchronous and Synchronous Operation) • Parallel Input and Output Ports • Four Channel Counter/Timer (Z80-CTC) • Software Programmable Baud Rate Generator • No-Front Panel Required for Operation • 4 MHz Operation.

SD'S VERSAFLOPPY II

• IBM 3740 Compatible Soft Sectored Format for Single Density Drives • Operates with Single and Dual Sided Drives, Single or Double Density Drives and 5" & 8" Drives — in any combination of four simultaneously • Drive Select and Side Select Circuitry • S-100 Bus Compatible • Vectored Interrupt Operation Optional • Phase Locked Loop Data Recovery Circuit • Operates with Z80 CPU's • Uses FD1791-1 Controller Chip • Ther Versafloppy II incorporates all the possible features of a flexible disk drive controller into one board. Capable of handling four drives simultaneously, combinations of any variety are possible, such as 5" single sided, 8" dual density dual sided, 5" dual density single sided. Most popular drives are controlled directly with the Versafloppy II. The operating system for the Versafloppy II is the extremely powerful SDOS available for SD Systems. Diagnostic and control software available to complete your disk system.

KIT $335.95

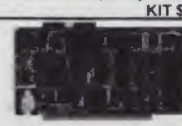
SD'S VDB-8024 VIDEO DISPLAY BOARD

The VDB-8024 features its own on-board Z80 microprocessor. This gives the capability of using software (included in ROM) to control functions and enhancements without interference with the computer's CPU. Included in the special features: 80 characters by 24 lines display, keyboard power and interface, composite and separate video output, 2K on-board RAM, a total of 256 available characters, full cursor control, forward and reverse scrolling, underlining, field reverse, field protect enhancements, programmable characters. KIT $329.95 A&T $389.95

SD'S "VERSAFLOPPY I" KIT

FEATURES: IBM 3740 soft sectored compatible, S-100 BNS Compatible for Z-80 or 8080. Controls up to 4 drives (single or double sided). Directly controls the following drives: Sugart SA400/450 Mini Floppy • Shugart SA800/850 Standard Floppy • PERSCI 70 and 277 • MFE 700/750 • CDC 9404/9406

$189.95



SD'S SBC-100 SINGLE BOARD COMPUTER

The SBC-100 provides a complete micro-computer on a single board! The Z80 microprocessor is used as the heart of the SBC-100. The SBC-100 meets all the requirements of a Z80 CPU board with the added features of I/O ports, counter/timer channels, on board RAM, provisions for PROM/ROM and a software programmable baud rate generator. S-100 Bus compatible, the SBC-100 features are: 8K bytes of available PROM, 1024 bytes on-board RAM, Serial I/O with both synchronous and asynchronous operation, Parallel I/O ports, Operational Vectored Interrupts, and Four Counter/Timer Channels. SD Monitor available for RS-232 and Video Terminals. Disk based system software also available.
SBC-100 KIT $249.95

TARBELL FLOPPY DISK INTERFACE

Compatible with Z80 & 8080. S-100 Bus. Uses CPM operating system. Plugs directly into your IMSAI or ALTAIR • Fastest transfer rate
KIT $190.00 Assembled & Tested $260.00

TARBELL CASSETTE INTERFACE

Plugs directly into your IMSAI or ALTAIR • Fastest transfer rate • Extremely reliable • Phase encoded • 4 extra status & control lines
KIT $99.95

SSM

CB2 Z-80 CPU Board Kit $186.00

Operates at 2 MHZ or 4 MHZ by DIP switch selection and includes two sockets for 2716 or 2732 EPROMs or TMS 4016 2K RAMs. Jumper options generate the new IEEE S-100 signals.

SSM

VB3 80 Character Video Board Kit $299.95

• 80 char. per line, up to 51 lines • Graphics up to 160 × 204 matrix • Up to 256 user defined symbols (optional EPROM) • Composite video

PB1 2708/2716 Programmer & 4K/8K EPROMBoard Kit SSM $124.00

• S-100 bus • 2 separate programming sockets for 2708 or 2716 (5V) EPROMs • Programming voltage generated on board — no need for an external power supply • Software control of 2708/2716 programming selection • LED indicator for programming mode and an on-off switch for programming voltage • 4 sockets for 4K of 2708 or 8K of 2716 EPROMs • Unused EPROM sockets do not enable data bus drive so the board is never committed to the full 4K or 8K of memory • Jumper selectable wait states (0-4) for fast or slow EPROMS

Z80 STARTER KIT

Kit: $279.95 Assembled & Tested $349.95

SD System's Z80 Starter Kit enables the novice to build a complete microcomputer on a single board. Featuring the powerful Z80 microprocessor, the Z80 Starter Kit features • Keyboard and Display • Audio Interface • PROM Programmer • Expansion and Wire Wrap Area • On Board RAM • 4 Channel Counter/Timer • Z-BUG Monitor in PROM • I/O Ports.

NOVATION CAT ACOUSTIC COUPLER/MODEM

$189.00

Let your computer communicate with other computers. Bell Systems 103 compatible 300 baud, answer or originate.

SYM-1

Reg. $269.00
NOW $219.00

- KIM-1 Compatible
- 4K ROM Monitor
- 1K Bytes 2114 RAM
- 65K Memory Expansion
- User EPROM 2716

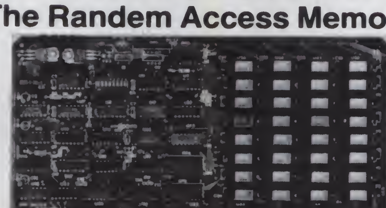
IO4 2 Parallel/2 Serial I/O Board Kit $126.00

• S-100 bus • 2 serial I/O ports (2 in & 2 out) • Independent baud-rate selection from 55 to 9600 baud • Regulated +5V, +12V & −12V outputs provided on both serial headers • 2 latched parallel I/O ports (2 in & 2 out) • Independent DIP switches for setting address • Interrupt capability provided for on serial and parallel I/O ports • +8V @ 0.95A, +16V @ 0.6A, & −16V @ 80mA typical SSM

VB1B VIDEO BOARD KIT $124.00

• S-100 bus • 64 or 32 characters per line (DIP switch selectable), 16 lines • Graphics 128×48 matrix • Upper case, lower case, Greek characters, symbols and numbers • 7×9 dot character matrix • Black-on-white or white-on-black • Timing 60Hz vertical rate, 16.2KHz horizontal rate, Crystal 12.44MHz • Parallel and composite video output (US TV signals), separate video, horizontal and vertical sync SSM

SD'S EXPANDORAM II
The Randem Access Memory



- S-100 Bus Compatible
- Up to 4Mhz Operation
- Expandable Memory from 16K to 256K
- DIP Switch Selectable Boundaries
- Uses 16K (4116) or 64K (4164) Memory Devices
- Page Mode Operation Allows up to 8 Memory Boards on Bus
- Operates with Z80 CPU's
- Phantom Output Disable
- Invisible Refresh (Synchronized with Wait States)

SD Systems' ExpandoRAM II is a dynamic RAM board with capacities from 16K bytes (4116) to 256K bytes (4164). It operates on the industry S-100 Bus. The design allows 8 boards to operate from the same S-100 Bus. The ExpandoRAM II is compatible with most S-100 CPU's based on the Z80 microprocessor.

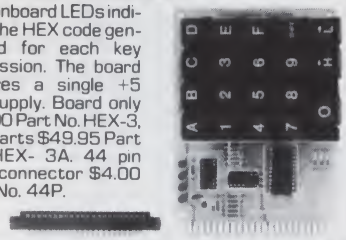
EXPANDORAM II KIT

16K	$295.95
32K	369.95
48K	444.95
64K	519.95

Radio Hut Radio Hut Radio Hut Radio Hut

HEX ENCODED KEYBOARD

Four onboard LEDs indicate the HEX code generated for each key depression. The board requires a single +5 volt supply. Board only $15.00 Part No. HEX-3, with parts $49.95 Part No. HEX- 3A. 44 pin edge connector $4.00 Part No. 44P.



ASCII TO CORRESPONDENCE CODE CONVERTER

This bidirectional board is a direct replacement for the board inside the Trendata 1000 terminal. The on board connector provides RS-232 serial in and out. Sold only as an assembled and tested unit for $249.95. Part No. TA 1000C

T.V. INTERFACE

• Converts video to AM modulated RF, Channels 2 or 3. So powerful almost no tuning is required. On board regulated power supply makes this extremely stable. Rated very highly in Doctor Dobbs' Journal. Recommended by Apple • Power required is 12 volts AC C.T., or +5 volts DC • Board only $7.60 part No. 107, with parts $13.50 Part No. 107A

TAPE INTERFACE

• Converts a low cost tape recorder to a digital recorder • Works up to 1200 baud • Digital in and out are TTL-serial • Output of board connects to mic. in of recorder • Earphone of recorder connects to input on board • No coils • Requires +5 volts, low power drain • Board only $7.60 Part No. 111, with parts $29.95 Part No. 111A

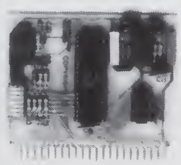
ASCII KEYBOARD

53 Keys popular ASR-33 format • Rugged G-10 P.C. Board • Tri-mode MOS encoding • Two-Key Rollover • MOS/DTL/TTL Compatible • Upper Case lockout • Data and Strobe inversion option • Three User Definable Keys • Low contact bounce • Selectable Parity • Custom Keycaps • George Risk Model 753. Requires +5, -12 volts. $59.95 Kit.

T.V. TYPEWRITER

• Stand alone TVT • 32 char/line, 16 lines, modifications for 64 char/line included • Parallel ASCII (TTL) input • Video output • 1K on board memory • Output for computer controlled curser • Auto scroll • Non-destructive curser • Curser inputs: up, down, left, right, home, EOL, EOS • Scroll up, down • Requires +5 volts at 1.5 amps, and -12 volts at 30 mA • All 7400, TTL chips • Char. gen. 2513 • Upper case only • Board only $39.00 Part No. 106, with parts $145.00 Part No. 106A

UART & BAUD RATE GENERATOR



• Converts serial to parallel and parallel to serial • Low cost on board baud rate generator • Baud rates: 110, 150, 300, 600, 1200, and 2400 • Low power drain +5 volts and -12 volts required • TTL compatible • All characters contain a start bit, 5 to 8 data bits, 1 or 2 stop bits, and either odd or even parity. • All connections go to a 44 pin gold plated edge connector • Board only $12.00 Part No. 101, with parts $35.00 Part No. 101A, 44 pin edge connector $4.00 Part No. 44P

ASCII KEYBOARD

TTL & DTL compatible • Full 67 key array • Full 128 character ASCII output • Positive logic with outputs resting low • Data Strobe • Five user-definable spare keys • Standard 22 pin dual card edge connector • Requires +5VDC, 325 mA. Assembled & Tested. Cherry Pro Part No. P70-05AB. $119.95.

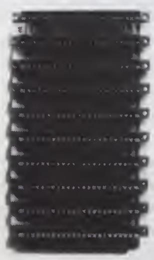
SOROC IQ 120

Upper/lower case display • Numeric keypad & cursor keys • Protected fields, ½ intensity display • RS 232 interface & aux. port. IQ120—$799.95 • IQ140 Detachable keyboard—$1199.95

MODEM

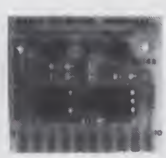
• Type 103 • Full or half duplex • Works up to 300 baud • Originate or Answer • Serial TTL input and output • connect 8 Ω speaker and crystal mic. directly to board • Requires +5 volts • Board only $7.60 Part No. 109, with parts $29.95 Part No. 109A.

44 BUS MOTHER BOARD



Has provisions for ten 44 pin (.156) connectors, spaced 3/4 of an inch apart. Pin 20 is connected to X, and 22 is connected to Z for power and ground. All the other pins are connected in parallel. This board also has provisions for bypass capacitors. Board cost $15.00 Part No. 102. Connectors $3.00 each Part No. 44WP.

RS-232/20mA INTERFACE



This board has two passive, opto-isolated circuits. One converts RS-232 to 20mA, the other converts 20mA to RS-232. All connections go to a 10 pin edge connector. Requires +12 and -12 volts. Board only $9.95, part no. 7901, with parts $14.95 Part No. 7901A.

COMPRINT PRINTER

Printing Characteristics: 225 characters/second (170 lines/minute) throughput • 9 horizontal x 12 vertical matrix • 96 ASCII character set with upper and true lower case • 80 characters/line • 5.8 lines/inch
Buffer Memory: standard 256 bytes; • optional; 2,048 bytes (buffer memory option designated as Model 912-2K), add $149.95.
Paper Requirements: electrosensitive type (aluminum coated) • 8-1/2 inch width • 3.7 inch max. (300 ft.) roll diameter.
Model 912-S Interfacing: serial interface RS232 and 20 mA current loop • BAUD rates 110, 150, 300, 600, 1200, 2400 and 4800 are strap selectable.
Model 912-P Interfacing: parallel interface, IEEE-488 and 8 bit parallel (strobe/ acknowledge). Model 912-S, Part No. CPIA, 32118, $579.95. Model 912-P, Part No. CPIA, 32117, $559.95.

RS-32/TTL INTERFACE

• Converts TTL to RS-232, and converts RS-232 to TTL • Two separate circuits • Requires -12 and +12 volts • All connections go to a 10 pin edge connector, kit $9.95 Part No. 232A 10 Pin edge connector $3.00 part No. 10P.

COMPUCOLOR II

With reg. keyboard
MOD3 8K $1595.95
MOD4 16K $1695.95
MOD 5 32K $1995.95
Now includes $250 more, worth of software and accessories with 101 key option add $134.95 with 117 key option add $179.95

DC POWER SUPPLY

• Board supplies a regulated +5 volts at 3 amps., +12, -12, and -5 volts at 1 amp. • Power required is 8 volts AC at 3 amps., and 24 volts AC C.T. at 1.5 amps. • Board only $12.50 Part No. 6085, with parts excluding transformers $42.50 Part No. 6085A

To Order: Mention part no. description, and price. In USA shipping paid by us for orders accompanied by check or money order. We accept C.O.D. orders in the U. S. only, or a VISA or Master Charge no., expiration date, signature, phone no., shipping charges will be added. CA residents add 6.5% for tax. Outside USA add 10% for air mail postage and handling. Payment must be in U. S. dollars. Dealer inquiries invited. 24 hour order line (408) 448-0800 ✓47

Send for FREE Catalog . . . a big self-addressed envelope with 41¢ postage gets it fastest!

ELECTRONIC SYSTEMS Dept. KB P. O. Box 21638, San Jose, CA USA 95151

apple II Or APPLE II PLUS

16K $975.95, Extra 16K E.S. RAM installed $74.95, extra 32K E.S. RAM installed $148.95.

PET COMPUTER

With 16K & monitor - $ 795. Dual Disk Drive - $10 95

PARALLEL TRIAC OUTPUT BOARD FOR APPLE II

This board has 8 triacs capable of switching 110 volt 6 amp loads (660 watts per channel) or a total of 5280 watts. Board only $15.00 Part No. 210, with parts $119.95 Part No. 210A.

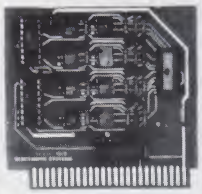
D.C. HAYES MICROMODEM

Fully S-100 bus compatible including 16-bit machines and 4 MHz processors. • Two software selectable Baud rates—300 Baud and a jumper selectable speed from 45 to 300 Baud. (110 standard). Supports originate and answer modes. • Direct-connect Microcoupler. This FCC-registered device provides direct access into your local telephone system, with none of the losses or distortions associated with acoustic couplers and without a telephone company supplied data access arrangement. • Auto-Answer/Auto-Call. The MICROMODEM 100 can automatically answer the phone and receive input; it can also dial a number automatically. • Automatic Reset and Disconnect. • Software compatible with the D.C. Hayes Associates 80-103A Data Communications Adapter. Micromodem-DCHA32625—$379.95

APPLE II HOBBY/ PROTOTYPING CARD

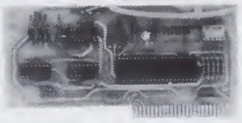
Part No. 7907 $14.95

OPTO-ISOLATED PARALLEL INPUT BOARD FOR APPLE II



There are 8 inputs that can be driven from TTL logic or any 5 volt source. The circuit board can be plugged into any of the 8 sockets of your Apple II. It has a 16 pin socket for standard dip ribbon cable connection. Board only $15.00. Part No. 120, with parts $69.95. Part No. 120A.

APPLE II SERIAL I/O INTERFACE

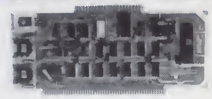


Baud rate is continuously adjustable from 0 to 30,000 • Plugs into any peripheral connector • Low current drain. RS-232 input and output • On board switch selectable 5 to 8 data bits, 1 or 2 stop bits, and parity or no parity either odd or even • Jumper selectable address • SOFTWARE • Input and Output routine from monitor or BASIC to teletype or other serial printer • Program for using an Apple II for a video or an intelligent terminal. Also can output in correspondence code to interface with some selectrics. • Also watches DTR • Board only $15.00 Part No. 2, with parts $42.00 Part No. 2A, assembled $62.00 Part No. 2C

REAL TIME 100,000 DAY CLOCK

MT. HARDWARE Double the utility of your *S-100 bus* computer with a real-time clock that keeps time in 100µS increments for over 273 years. Program events for the entire period with real time interrupts...without derailing the system. Maintain a log of computer usage, time and date transaction printouts, call up lists...virtually any activity where time is a factor. On-board battery backup. MHPX004—$249.95

TIDMA

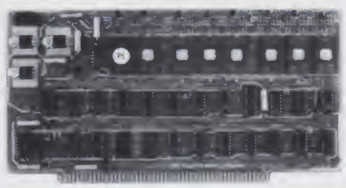


Tape Interface Direct Memory Access • Record and play programs without bootstrap loader (no prom) has FSK encoder/decoder for direct connections to low cost recorder at 1200 baud rate, and direct connections for inputs and outputs to a digital recorder at any baud rate • S-100 bus compatible • Board only $35.00 Part No. 112, with parts $110.00 Part No. 112A.

SYSTEM MONITOR

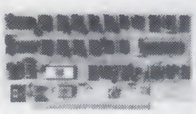
8080, 8085, or Z-80 System monitor for use with the TIDMA board. There is no need for the front panel. Complete with documentation $12.95.

8K EPROM PICEON



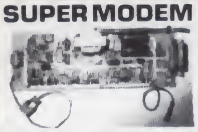
• Programs 2708's address relocation of each 4K of memory to any 4K boundary • Power on jump and reset jump option for "turnkey" systems and computers without a front panel • Program saver software in 1 2708 EPROM $25. Bare board $35 including custom coil, board with parts but no EPROMS $139, with 4 EPROMS $179, with 8 EPROMS $219.

VIDEO TERMINAL



16 lines, 64 columns • Upper and lower case • 5x7 dot matrix • Serial RS-232 in and out with TTL parallel keyboard input • On board baud rate generator 75, 110, 150, 300, 600, & 1200 jumper selectable • Memory 1024 characters (7-21L02) • Video processor chip SFF96364 by Neculonic • Control characters (CR, LF, →, ←, ↑, ↓, non destructive cursor, CS, home, CL • White characters on black background or vice-versa • With the addition of a keyboard, video monitor or TV set with TV interface (part no. 107A) and power supply this is a complete stand alone terminal • also S-100 compatible • requires +16, & −16 VDC at 100mA, and 8VDC at 1A. Part No. 1000A $199.95 kit.

SUPER MODEM



Orignate, RS-232 and 20 mA compatable, Full duplex, and half duplex, direct connect or acoustic coupled, on board power supply, carrier detect light, DB25 plug, 300 BAUD, Type 103 compatable frequencies, Bare board Part No. 2000, $19.95, Kit Part No. 2000A, $99.95.

RS-232/TTY INTERFACE

This board has two active circuits, one converts RS-232 to 20 mA, the other converts 20 mA to RS-232. Requires +12 and -12 volts. $9.95 Part No. 600A Kit.

SERIAL I/O

Four Serial I/O RS-232 ports. S-100 Bus, Software or jumper selectable baud rate (110, 300, 600, 1200, 2400, 4800, 9600, 19.2K), on board Xtal baud rate generator, Addressing, switch selectable, Parity or no parity (odd or even) switch selectable, 1 or 2 stop bits, 5 to 8 bits/character. Board only $29.95, Part No. 7908. With parts (kit) $199.95, Part No. 7908A.

16K EPROM

Uses 2708 EPROMS, memory speed selection provided, addressable anywhere in 65K of memory, can be shadowed in 4K increments. Board only $24.95 part no. 7902, with parts less EPROMs $49.95 part no. 7902A.

WAMECO PRODUCTS

With ELECTRONIC SYSTEMS parts

FDC-1 FLOPPY CONTROLLER BOARD will drive shugart, pertek, remex 5" & 8" drives up to 8 drives, on board PROM with power boot up, will operate with CPM (not included). PCBD $42.95

FPB-1 Front Panel. (Finally) IMSAI size hex displays. Byte or instruction single step. PCBD $42.95

MEM-1A 8Kx8 fully buffered, S-100, uses 2102 type RAMS. PCBD .... $24.95, $168 Kit

QMB-12 MOTHER BOARD, 13 slot, terminated, S-100 board only .... $34.95 $89.95 Kit

CPU-1 8080A Processor board S-100 with 8 level vector interrupt PCBD .. $25.95 $89.95 Kit

RTC-1 Realtime clock board. Two independent interrupts. Software programmable. PCBD .... $25.95, $60.95 Kit

EPM-1 1702A 4K EPROM card PCBD .... $25.95 $49.95 with parts less EPROMS

EPM-2 2708/2716 16K/32K EPROM card PCBD .... $24.95 $49.95 with parts less EPROMS

QMB-9 MOTHER BOARD. Short Version of QMB-12. 9 Slots PCBD .... $30.95 $67.95 Kit

MEM-2 16Kx8 Fully Buffered 2114 Board PCBD .... $25.95, $269.95 Kit

S-100 BUS ACTIVE TERMINATOR

Board only $14.95 Part No. 900, with parts $24.95 Part No. 900A

To Order: Mention part no. description, and price. In USA shipping paid by us for orders accompanied by check or money order. We accept C.O.D. orders in the U.S. only, or a VISA or Master Charge no., expiration date, signature, phone no., shipping charges will be added. CA residents add 6.5% for tax. Outside USA add 10% for air mail postage and handling. Payment must be in U.S. dollars. Dealer inquiries invited. 24 hour order line (408) 448-0800

Send for FREE Catalog . . . a big self-addressed envelope with 41¢ postage gets it fastest!

ELECTRONIC SYSTEMS Dept. KB P. O. Box 21638, San Jose, CA USA 95151

MICROCOMPUTING T.M. LIST OF ADVERTISERS

*Reader Service inquiries not honored. Please contact advertiser directly.

GIMIX inc.

1337 WEST 37th PLACE • CHICAGO, ILLINOIS 60609 • (312) 927-5510 • TWX 910-221-4055

We have a wide variety of SS50 bus items available. They are compatible with existing SWTP, MSI, and other manufacturer's products and the DIP-switch features on GIMIX systems allow use of a broad range of existing software. The below items are normally in stock except for systems which are customized to your needs. If you don't see what you want, please ask. We believe in service. All our boards and systems are assembled, burnt-in, and tested.

HANDLING CHARGES
U.S. On Orders Under $200.00
Add $5.00

32K 6800 SYSTEMS: Include 6800 CPU Board, 32K RAM Board, Mainframe Cabinet, Mother Board, Power Supply, Fan, Choice of One 30 Pin I/O Card and Cable, Choice of Two Filler Plates or Two Disk Regulator Cards, Binder, Documentation, and Manuals	$1,594.59
16K 6800 SYSTEMS: As Above, With 16K Version of 32K RAM Board	1,374.49
For 6800 CPU with 6840 Timer Option ... Add	64.00
VIDEO BASED SYSTEMS: When Ordered With Either of the Above. Video Based Systems Require Only A Parallel Keyboard and A Video Monitor and Are Faster, Lower In Cost, and Give You More Versatility and Features For Your Money Than Most Terminals.	
64 x 16 VIDEO BOARD PLUS GMXBUG V.2.0 ... Add	200.00
80 x 24 VIDEO BOARD (Without RAM Set) PLUS GMXBUG V.3.0 ... Add	400.00
80 x 24 VIDEO BOARD (Full Features) PLUS GMXBUG V.3.0 PLUS THE MANUAL AND DISK FOR PROGRAMMING USER DEFINED CHARACTER SETS ... Add	460.00
12 INCH VIDEO MONITOR (60Hz Only), Keyboard and Enclosure, Cables and Connectors	350.00
6800/6809 MAINFRAME (Includes Chassis, Power Supply, Switches, Fan and Mother Board Plus Baud Rate Generator on Mother Board)	828.19
6800/6809 MOTHER BOARD ONLY	294.09
DISK REGULATOR BOARD	18.91
FILLER PLATES	14.92
6800 CPU BOARD — WITHOUT TIMER	224.03
6800 CPU BOARD WITH 6840 TIMER PACKAGE	288.06
MEMORY BOARDS	
32K STATIC RAM, SOCKETED, WITH EXTENDED ADDRESSING	548.15
24K VERSION OF ABOVE (SOCKETED FOR 32K)	438.14
16K VERSION OF ABOVE (SOCKETED FOR 32K)	328.12
16K GHOSTable STATIC RAM	368.16
8K 2708 EPROM BOARD	98.34
I.O. BOARDS	
30 PIN SINGLE PORT SERIAL	88.41
30 PIN 2 PORT PARALLEL	88.42
50 PIN 8 PORT SERIAL (Available Feb. '80)	288.40
50 PIN 8 PORT SERIAL WITH BAUD RATE GENERATORS	318.46
50 PIN 8 PORT PARALLEL	198.45
CABLE SETS	18.95
VIDEO BOARDS	
64 or 32 x 16	$198.71
80 x 24 WITHOUT PROGRAMMABLE CHARACTER SET RAM	398.74
80 x 24 ALL FEATURES	458.76

SOFTWARE	
GMXBUG MONITORS:	
2K VERSION 1 — ON 2 2708s	68.54
3K VERSION 2 — ON 3 2708s	88.55
3K VERSION 3 — ON 3 2708s	98.56
VERSION 1 MANUAL ONLY WITH SOURCE LISTINGS — NO PROMS	28.51
VERSION 2 MANUAL ONLY WITH SOURCE LISTINGS — NO PROMS	28.52
VERSION 3 MANUAL ONLY WITH SOURCE LISTINGS — NO PROMS	38.53
PROGRAM TO CREATE USER DESIGNED CHARACTERS — MANUAL & DISK	20.61
POWER CONTROL BOARDS	
Contact GIMIX for Pricing on Complete Turnkey GHOST Systems with Real Time Software.	
RELAY DRIVER PACKAGE Assembled and Tested With 31 Relays, Bracket and Transformer	848.08
RELAY DRIVER BOARD ONLY	448.86
BRACKET	38.21
TRANSFORMER	14.24
G.E. RR8 20 AMP RELAY	11.50
OPTO ISOLATED INPUT BOARD (34 SWITCH CLOSURES)	348.85
16-BUTTON REMOTE 2-WIRE KEYBOARD SYSTEM KEYBOARD	98.83
TONE RECEIVER BOARD	248.84
35 BUTTON SERIAL KEYBOARDS — Wood Laminate, Black, White or Walnut	118.82
35 BUTTON SERIAL KEYBOARDS — Clear or Smoke Acrylic	128.82
FACTORY PRIME CHIPS	
4044s ... 450 ns	5.90
4044s ... 250 ns	6.90
2114Ls ... 450 ns	5.90
2114Ls ... 200 ns	6.90
2708s ... 450 ns	8.90
These are factory prime chips from same shipments we use in our professional quality boards.	
EXPORT NOTES	
FOR 50Hz 230V C.V. POWER SUPPLY ... Add	30.00
80 x 24 VIDEO BOARDS — Specify Format (No Added Charge)	
On orders under $250.00 for a Single Board, or Chips, please Add $30.00 Handling and we will ship Air Mail Prepaid. On all other orders we will ship via Emery Air Freight Collect, and we will charge no handling. All orders must be prepaid in U.S. Funds. Please note that foreign checks have been taking about eight weeks for collection, so we would advise wiring money or checks drawn on a bank account in the U.S. Our bank is the Continental Illinois National Bank of Chicago, account #73-32033. Visa or Master Charge also accepted.	

NOTE: OUR PRICE IS OUR CATALOG NUMBER, PLEASE BE SURE TO PRICE OUT COMPLETELY.

READER SERVICE

Please help us to bring you a better magazine—by answering these questions:

I. What is your age?
- ☐ 1. under 18
- ☐ 2. 18-22
- ☐ 3. 23-40
- ☐ 4. 41-60
- ☐ 5. over 60

II. What are your primary applications for microcomputing (check only two)?
- ☐ A. Business
- ☐ B. Games
- ☐ C. Home
- ☐ D. Education
- ☐ E. Scientific
- ☐ F. Control
- ☐ G. Music

III. What other magazines do you subscribe to (check all that apply)?
- ☐ 1. 80 Microcomputing
- ☐ 2. 73 Magazine
- ☐ 3. Byte
- ☐ 4. Creative Computing
- ☐ 5. Interface Age
- ☐ 6. Personal Computing

IV. How much have you spent on hardware?
- ☐ A. less than $500
- ☐ B. $500-1,000
- ☐ C. $1,000-2,000
- ☐ D. $2,000-4,000
- ☐ E. $4,000-6,000
- ☐ F. more than $6,000

V. How much have you spent on software?
- ☐ 1. less than $100
- ☐ 2. $100-250
- ☐ 3. $250-500
- ☐ 4. $500-1,000
- ☐ 5. more than $1,000

VI. What do you plan to buy during the next year (check all that apply)?
- ☐ A. Complete system
- ☐ B. Disk system
- ☐ C. Impact printer
- ☐ D. Dot matrix printer

VII. What system do you own and use most (check only one)?
- ☐ 1. Radio Shack
- ☐ 2. Commodore
- ☐ 3. Apple
- ☐ 4. Atari
- ☐ 5. Heath
- ☐ 6. SWTP
- ☐ 7. Exidy
- ☐ 8. North Star
- ☐ 9. Compucolor
- ☐ 10. other

VIII. How many disk drives do you own?
- ☐ A. none
- ☐ B. 1
- ☐ C. 2
- ☐ D. 3
- ☐ E. 4
- ☐ F. more than 4

IX. How many people read your copy of Kilobaud Microcomputing?
- ☐ 1. 1
- ☐ 2. 2
- ☐ 3. 3
- ☐ 4. 4 or more

X. If you are not a subscriber please circle number 500.

Kilobaud MICROCOMPUTING • April 1980

Reader Service: Return this card to receive full information on the products advertised in this issue. Refer to the ad. You will find numbers near the logo of each advertiser. Each represents the advertiser's individual Reader Service number. Circle the corresponding numbers on one of the cards on this page, include your name, address & zip, and drop in a mailbox. In 4–6 weeks you'll hear from the advertiser directly. *This card is valid until July 31, 1980.*

1	6	11	16	21	126	131	136	141	146	251	256	261	266	271	376	381	386	391	396
2	7	12	17	22	127	132	137	142	147	252	257	262	267	272	377	382	387	392	397
3	8	13	18	23	128	133	138	143	148	253	258	263	268	273	378	383	388	393	398
4	9	14	19	24	129	134	139	144	149	254	259	264	269	274	379	384	389	394	399
5	10	15	20	25	130	135	140	145	150	255	260	265	270	275	380	385	390	395	400
26	31	36	41	46	151	156	161	166	171	276	281	286	291	296	401	406	411	416	421
27	32	37	42	47	152	157	162	167	172	277	282	287	292	297	402	407	412	417	422
28	33	38	43	48	153	158	163	168	173	278	283	288	293	298	403	408	413	418	423
29	34	39	44	49	154	159	164	169	174	279	284	289	294	299	404	409	414	419	424
30	35	40	45	50	155	160	165	170	175	280	285	290	295	300	405	410	415	420	425
51	56	61	66	71	176	181	186	191	196	301	306	311	316	321	426	431	436	441	446
52	57	62	67	72	177	182	187	192	197	302	307	312	317	322	427	432	437	442	447
53	58	63	68	73	178	183	188	193	198	303	308	313	318	323	428	433	438	443	448
54	59	64	69	74	179	184	189	194	199	304	309	314	319	324	429	434	439	444	449
55	60	65	70	75	180	185	190	195	200	305	310	315	320	325	430	435	440	445	450
76	81	86	91	96	201	206	211	216	221	326	331	336	341	346	451	456	461	466	471
77	82	87	92	97	202	207	212	217	222	327	332	337	342	347	452	457	462	467	472
78	83	88	93	98	203	208	213	218	223	328	333	338	343	348	453	458	463	468	473
79	84	89	94	99	204	209	214	219	224	329	334	339	344	349	454	459	464	469	474
80	85	90	95	100	205	210	215	220	225	330	335	340	345	350	455	460	465	470	475
101	106	111	116	121	226	231	236	241	246	351	356	361	366	371	476	481	486	491	496
102	107	112	117	122	227	232	237	242	247	352	357	362	367	372	477	482	487	492	497
103	108	113	118	123	228	233	238	243	248	353	358	363	368	373	478	483	488	493	498
104	109	114	119	124	229	234	239	244	249	354	359	364	369	374	479	484	489	494	499
105	110	115	120	125	230	235	240	245	250	355	360	365	370	375	480	485	490	495	500

Name ______________________

Address ______________________

City ______________ State ________ Zip ________

BOOKS

Please send me the following Microcomputing products:

Qty.	Catalog #	Title	Unit Price	Total

Add $1 shipping/handling ________

Total ________

Please allow 4-6 weeks for delivery.

Enclosed $__________ ☐ Check ☐ M.O.

Bill: ☐ AE ☐ MC ☐ Visa

Card # ______________ Exp. date ________

Name ______________________

Address ______________________

City ______________________

State ______________ Zip ______________

4/80

SUBSCRIPTION

MICROCOMPUTING subscribers save $12 off the newsstand price.

- ☐ New subscription ☐ Renewal
- ☐ 1 year—$18
- ☐ 2 years—$30
- ☐ 3 years—$45

Enclosed $__________ ☐ Check ☐ M.O.

Bill: ☐ MC ☐ Visa ☐ AE ☐ me

Card # ______________ Exp. date ________

Name ______________________

Address ______________________

City ______________ State ________ Zip ________

Canada: same as US, in US funds.
Other foreign: $23 one year only, US funds.

304B9

kilobaud MICROCOMPUTING

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 217 CLINTON IA 52735

POSTAGE WILL BE PAID BY ADDRESSEE

kilobaud MICROCOMPUTING T.M.

POB 2741
Clinton IA 52735

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 1024 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

kilobaud MICROCOMPUTING T.M.

Subscription Dept.
POB 997
Farmingdale NY 11737

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 1024 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

kilobaud MICROCOMPUTING T.M.

Peterborough NH 03458

Att. Mail Order

Expand Your Business Four Times This Year

Expanding a business four times in one year may be easier said than done except for your data processing needs. The new MSI Multi-User Computer System is designed to grow with your business by giving you the capability to add up to four user terminals which can be operated simultaneously.

At the heart of the system is the powerful MSI 6800 computer combined with MSI Multi-User BASIC and the new SDOS operating system.

Utilizing the new MSI 16K Static RAM modules expands processor memory to 152K of RAM. Each of the four users commands 32K of processor RAM as well as simultaneous access to the data base.

MSI Multi-User BASIC can be used with any MSI disk memory system including the HD-8/R 10 megabyte and HD-76/R 76 megabyte fixed/removable hard disk systems.

Our new Multi-User Computer System also features a software controllable printer option which allows you to use both a "daisy wheel" word processor for high quality document preparation, and a dot matrix printer for high speed production.

MSI currently offers nine different computer systems for business, industrial, scientific, educational and personal applications. And software to make your data processing more effective includes a Business Accounting Package with accounts receivable, accounts payable, general ledger, and inventory control.

If your business is expanding and you would like to know how an MSI computer system can help you make it more profitable, call or write Midwest Scientific Instruments, 220 W. Cedar, Olathe, Kansas 66061, (913) 764-3273, TWX 910 749 6403 (MSI OLAT), TELEX 42525 (MSI A OLAT)

144

Small Computers For Big Jobs

MSI Midwest Scientific Instruments